Image Retrieval

Deep Learning in Practice - M2 MVA

Tong ZHAO (tong.zhao@eleves.enpc.fr)

1 Creating a Network with the AlexNet Architecture

Q1. Each row of the matrix represents the extracted feature of the corresponding image. It is the output of the last second fully-connected layer.

Q2. The dimension of the feature is 4096, which corresponds to the output feature dimension of the layer *fc7*.

In order to extract these features, we first load a pretrained neutral network and then we pass the images to the network. After the forward process is finished, we simply take the output of the selected layer as the extracted feature.

Q3. By using the t-SNE visualization, we can embed the high dimensional features in a 2D space, which allows us to analysis the distribution of each class.

We can observe that the cluster is not good enough. Some classes don't have a cluster in the space, instead the features are dsitributed everywhere in the space.

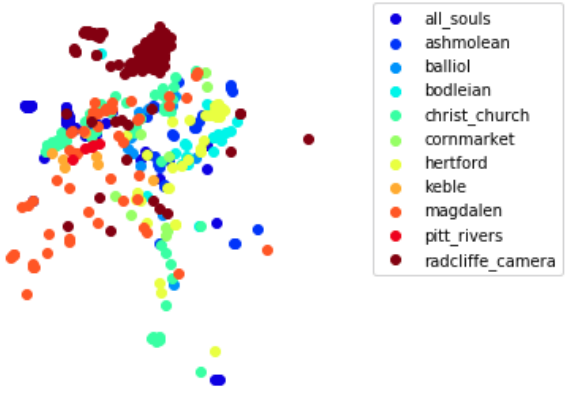

Figure 1: Pretrained AlexNet on ImageNet

Radcliffe_ camera, *bodleian*, *keble* have a good clustering whereas the ones of *all_ souls*, *hertford*, *magdalen*, *christ_ church* and *ashmolean* are not good enough.

2 Finetuning the Created Network on the Landmarks Dataset

Q4. The result should be better since most of the images in the Landmarks dataset are buildings, which consist with our test dataset. We hope that the features can focus more on the architectures

thus we could have a better representation.

The network need to be changed slightly in order to adapt to the new training dataset.

Q5. There are two reasons to change the last layer of the AlexNet. First of all, the number of classes in Landmarks dataset is 586 instead of 1000 in ImageNet. The second reason is that we can consider the last layer as a classifier which distributes a label to each image according to their features from the previous layer. Since the classes are changed, it is reasonable to have another classifier to achieve our goal.

Q6. We use the weights of AlexNet to initialize all layers except the last one. Then the last layer can be initialized randomly or by a more advanced method like Xavier initialization or Kaiming initialization.

Q6 bis. The t-SNE result is slightly better after finetuning. The best clustered class does not change so much while the other classes improve.

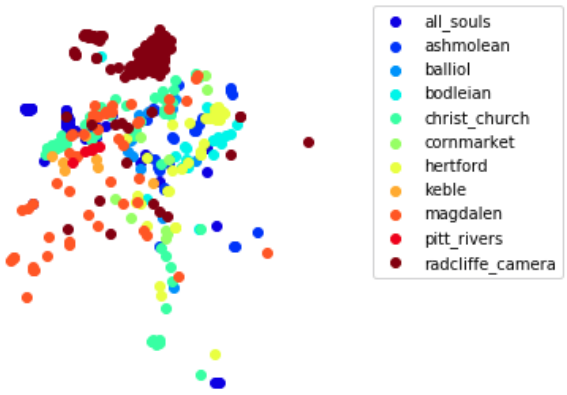

Figure 2: Finetuned AlexNet on Landmarks

Q7. The structure of AlexNet decides that the input image size should always be 224×224. If the size changes, we can't use the weights of all fully connected layers of the pretrained model.

However this resize operation has a bad effect since many images are distorted thus the features are biased.

3 Replacing Last Max Pooling Layer with GeM Layer

Q8. The feature dimension is changed this time since we discard all fully connected layer and use a GeM layer to extract a value per channel from the last convolutional layer. Given that the last convolutional layer has 256 channels, the feature dimension is also 256.

Q9. The size of the feature representation is crucial for image retrieval. It decides the power of the representations. A higher dimension can give us more information but it may leads to overfitting.

Q10. From the plot, we can see that the images from a same class become closer. It means that the features are more meaningful thus the distance gives a better summary of the similarity between two images.

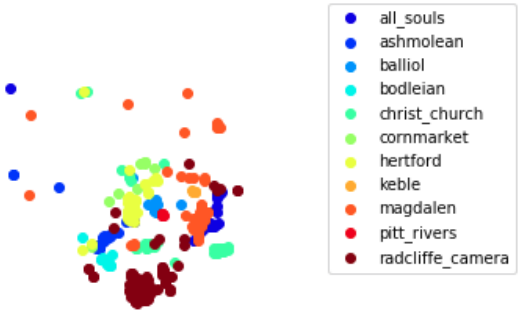

Figure 3: AlexNet with GeM Layer

Q11. The clusters of two similarly labeled images are closer in general. Taking the example of the *bodlenian* class and the *radcliffe_ camera* class, the distance between the two clusters is closer compared to the others.

4 ResNet18 Architecture with GeM Pooling

Q12. The generalized mean pooling layer takes a parameter p as input, which controls the smoothness of the pooling operator. If $p = 1$, it is equivalent to mean pooling and if $p \to \infty$, it is equivalent to max pooling. What's more, the operator is differentiable.

Q13. The GeM layer calculates a powered average for each channel of the input. Given an input I of n channels, the output f can be written as:

$$f_i = \Big(\frac{1}{|I_i|}\sum_{x \in I_i} x^p\Big)^{1/p}$$

Q14. Generally speaking, this model is better than the model 1c. The distance between different classes are larger and we can better observe the clusters.

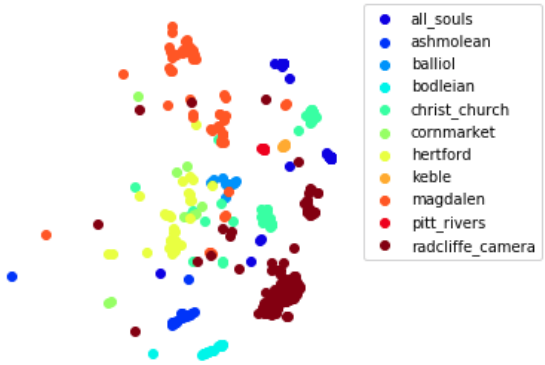

Figure 4: ResNet18 with GeM Layer

Q15. Compared to the finetuned AlexNet on Landmarks dataset, the clusters in 1d are better seperated and similar classes are closer.

5 PCA Whitening

Q16. We can observe that most of the unlabeled data are far from the labeled set, which is what we expect. However it is difficult to define a boundary, i.e. a threshold to decide if an image is labeled or unlabeled.

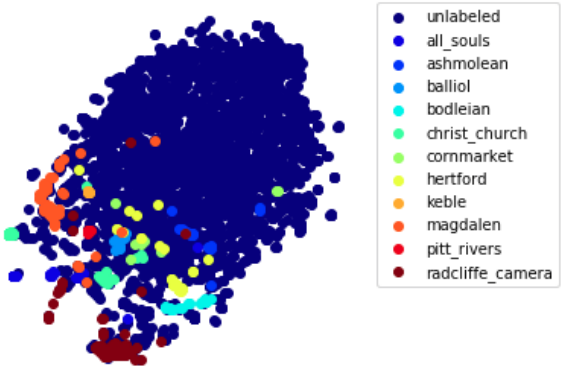

Figure 5: ResNet18 with PCA Whitening

Q17. The distribution of the unlabeled features are almost uniform in the whole space.

Q18. We can train a Siamese Network with triplet loss. Given a pair of images, the network decides if they belong to the same class or not. Thus if an image is not from any labeled class, it is a unlabeled image.

6 Finetuning on Landmarks for Retrieval

Q19. Compared with the plot obtained from the previous model, the current one seperates better the unlabeled data. The distances among clusters are also larger.

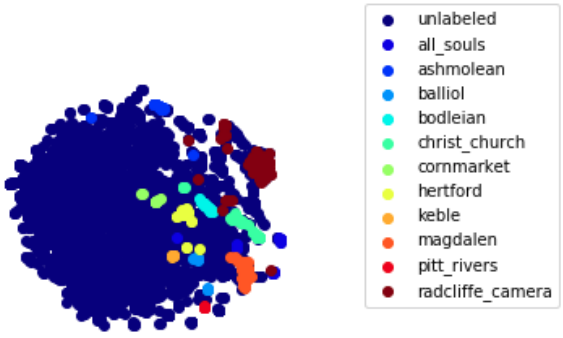

Figure 6: Finetuned ResNet18 with Triplet Loss

7 Data Augmentation and Multi-Resolution

Q20. On the chosen image (index = 15), the AP is improved from 55.24 to 58.36. In the candidate image set, many of them are not centered and the query image slightly slants to left. Thus the data augmentation help us find a better result.

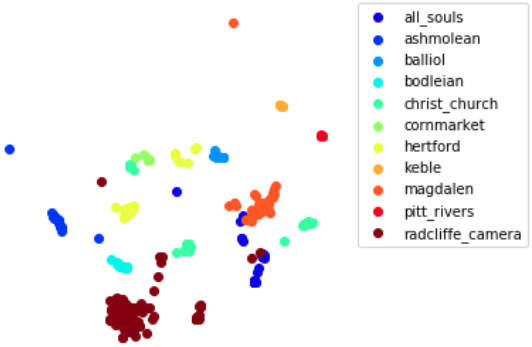

Figure 7: Finetuned ResNet18 with Data Augmentation and Multi-Resolution

Q21. The distances among clusters become even larger. However we should notice that some of them are farer away from their clusters, like the one in *magdalen*.

Q22. In terms of data augmentation, we can also use image flipping, image translation. In terms of pooling techniques, we can let the network find the best parameter p or we can use different p for each channel.

8 Improved Architecture

Q23. This we can observe that all the found matchings are correct. A larger architecture give the network a higher capacity to learn information from data. However we should also control it in order to avoid overfitting. It is a trade-off between this two situations.

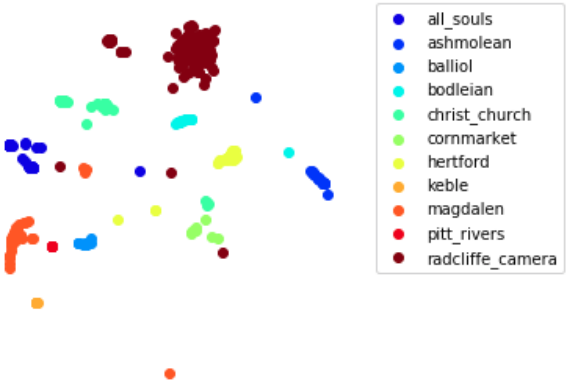

Figure 8: Finetuned ResNet50