2.4.3 VGG19

Extremely similar to VGG16, the original VGG19 network has 19 layers and is trained on more than a million images from the ImageNet database. The network can identify 1000 categories with input size of 224\*224. Figure X illustrates the structure of VGG19. It consists of five conventional layers and an output layer. Every conventional layer includes 2 or 4 conventional sub-layers, and there is one maxpool layer between every two conventional layers.



Figure 1 The structure of VGG19

In this paper, we also present a transfer learning solution to migrate the original VGG19 network to a relatively simplified map classifying network. To implement transfer learning, similar to VGG16, we loaded pre-trained weight, added a new output layer, and set the last few layers trainable during the training process.

4.3.3 VGG19

[Parameter setting]

Considering the extreme similarity between VGG16 and VGG19, we constructed the output layers and parameters just like VGG16.

[Experiment results]

Likewise, we investigated training/validation accuracy’s relationship with epochs, and trainable layer number.

Figure X shows the convergence process of VGG19 with 3 trainable layers. After 50 epochs, the accuracy and loss will converge and stay invariable. Considering the convergence speed of the network, we set the 20 epochs for all training instances.

Figure 2 The convergence process of VGG19

To see the trainable layers’ impact on the final accuracy, we conducted a series of experiment with different trainable layers’ number from 3 to 10. Figure X shows the convergence process of VGG19 with 3, 4, 5 trainable layers. Figure X shows the accuracy’s changing trend with the number of trainable layers.

Figure 3 The convergence process of VGG19 with 3, 4, 5 trainable layers.

|  |  |  |
| --- | --- | --- |
| Trainable layers | Training accuracy | Validation accuracy |
| 3 | 0.64 | 0.60 |
| 4 | 0.72 | 0.69 |
| 5 | 0.53 | 0.57 |
| 6 | 0.25 | 0.27 |
| 7 | 0.25 | 0.27 |
| 8 | 0.25 | 0.24 |
| 9 | 0.25 | 0.26 |
| 10 | 0.25 | 0.22 |

Table 1 Training and validation accuracy's relationship with trainable layers

Figure 4 The changing trend of accuracy with the number of trainable layers

Figure 5 shows the number of trainable layers’ impact on training time and trainable parameters. We can observe a strong correlation between time and parameters. Combined with Figure 4, we can also witness that final convergent accuracy is also correlated with trainable parameters. For trainable layers of 3, 4, and 5, the model can achieve higher accuracy with fewer trainable parameters. However, with increasing trainable parameters, the convergent accuracy instantly drops to random level.

Figure 5 Training time (seconds) and trainable parameter's relationship with the number of trainable layers

Figure 6 shows the convergent accuracy’s relationship with input data size. There is a valley from 1000 – 1200. We can observe: smaller dataset can achieve a relatively high accuracy, however, this could be overfitting. Larger dataset can also achieve higher accuracy in a good sense.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | |
| **# Trainable**  **layers**  **# Images**  **used** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| **400** | 0.20 | 0.80 | 0.80 | 0.38 | 0.23 | 0.23 | 0.20 | 0.20 | 0.20 |
| **1200** | 0.92 | 0.74 | 0.53 | 0.25 | 0.20 | 0.20 | 0.29 | 0.29 | 0.29 |
| **2000** | 0.71 | 0.33 | 0.87 | 0.20 | 0.33 | 0.33 | 0.33 | 0.20 | 0.33 |
| **2800** | 0.24 | 0.46 | 0.71 | 0.24 | 0.30 | 0.24 | 0.30 | 0.24 | 0.24 |
| **3600** | 0.81 | 0.27 | 0.23 | 0.24 | 0.27 | 0.26 | 0.24 | 0.23 | 0.23 |
| **4400** | 0.23 | 0.80 | 0.63 | 0.23 | 0.28 | 0.23 | 0.23 | 0.23 | 0.23 |
| **5200** | 0.74 | 0.66 | 0.76 | 0.25 | 0.23 | 0.23 | 0.26 | 0.26 | 0.23 |
| **6000** | 0.60 | 0.69 | 0.57 | 0.27 | 0.27 | 0.24 | 0.26 | 0.22 | 0.22 |

Table 2 The convergent accuracy's relationship with input data size

Figure 6 The convergent accuracy's relationship with input data size

[Discussion]

From Figure 4 we can observe that the accuracy reaches its peak at trainable layers of 4. The highest accuracy is 0.72, which is not ideal compared to the original network. However, with more trainable layers, the accuracy is lower. After 6 trainable layers, the convergent accuracy is 0.25, which is equal to random guessing. More parameters make the optimizer harder to achieve a better solution other than a local one, and we can observe this in Figure 3. For trainable layers of 3, it has a valley in the curve, which means the solution can easily jump out of the local optima. For 4, the accuracy oscillates around 0.65, which means the optimizer is effective. Nevertheless, after 5, especially 6, the accuracy hardly changes.

VGG19 has more parameters than VGG16, therefore, VGG19’s performance of transfer learning is accordingly worse than VGG16.