CIFAR-10 Classification using VGG,ResNet,GoogLeNet

Author:
Chi Zhang

November 10, 2022

Contents

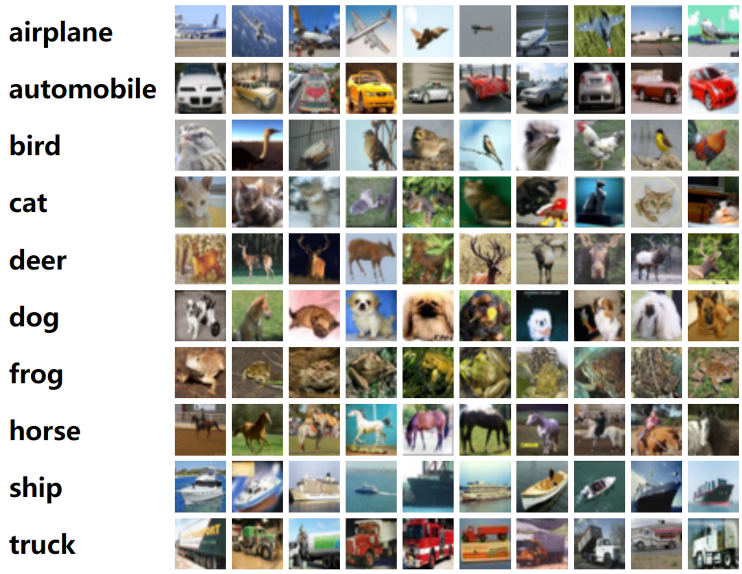


Figure 1: The Preview of Dataset

1 Introduction and Overview of Dataset

The CIFAR-10 dataset consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images.

The dataset is divided into five training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 5000 images from each class. The Fig 1 shows a preview of the dataset.

2 VGG

2.1 The Architecture of VGG

In Fig 2, we give the model construction diagram of VGG.

2.2 Features of VGG

1. Small convolution kernel : 3x3 convolution kernel
2. Small pooling kernel: 2x2 pooling kernel
3. Deeper layers with wider feature maps: Based on the first two points in addition to the convolution kernel focusing on expanding the number of channels and pooling focusing on narrowing the width and height, making the model architecturally deeper and wider while the computational effort slowly increases.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 2: The Architecture of VGG

2.3 Why is VGG designed in such a way

In VGG16, the authors consider that the perceptual field size obtained by a stack of two 3x3 convolutions is equivalent to a 5x5 convolution; while the perceptual field obtained by a stack of three 3x3 convolutions is equivalent to a 7x7 convolution. This reduces the parameters and increases the network depth on the one hand, and corresponds to more nonlinear mappings on the other hand, which can increase the fitting ability of the network.

2.4 My Choice

I choose the VGG-16 network, which is the class D in Fig 2.

2.5 Experiment

We choose the optimizer as SGD, set the learning rate to 0.01, use the cross-entropy loss function, and cut the training set into 167 batches with 300 as the training set batch size, and the number of epochs is 20.The device we use is Nvidia Geforce RTX 3050 Laptop GPU.

In Fig 3,We plot the training loss, training accuracy, and testing accuracy during model training. In Table 1, we give the GPU time and final accuracy for a single epoch versus all (20) epochs.

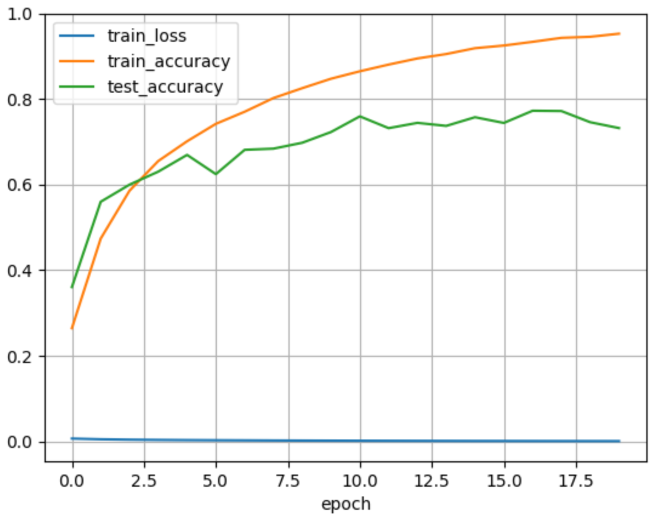


Figure 3: Training loss, training accuracy, and testing accuracy during model training

Epoch	**Computation time (ms)**	**Test accuracy (%)**
1	49216.24	43.00
20	1132012.75	73.23

Table 1: Computational efficiency and Test accuracy

3 ResNet

3.1 The Architecture of ResNet

In Fig 4, we give the model construction diagram of ResNet.

3.2 Features of ResNet

1. ResNet follows the complete 3x3 convolutional layer design of VGG
2. Put forward the concept of residual blocks.Fig 5 shows the structure of the residual block.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

:ures for ImageNet. Building blocks are shown in brackets (see also Fig. 5), with the numbers of block

Figure 4: The Architecture of ResNet

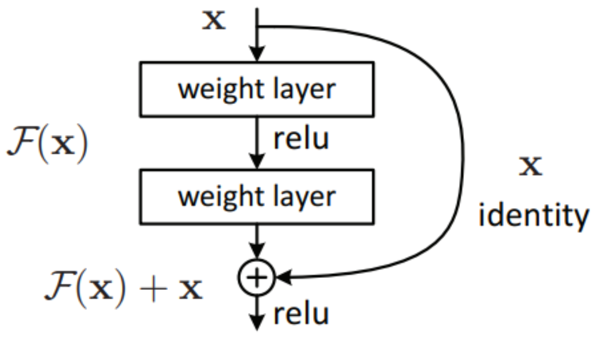


Figure 5: The Architecture of residual block

Epoch	**Computation time (ms)**	**Test accuracy (%)**
1	49216.24	51.40
20	395522.34	67.67
50	979699.44	73.63

Table 2: Computational efficiency and Test accuracy

3.3 Why is ResNet designed in such a way

There are two main design ideas for residual blocks, shortcut connections and constant mappings. Shortcut connections make residuals possible, while constant mappings make the network deeper.

3.4 My Choice

I choose the ResNet18 network, which is second column in Fig 4.

3.5 Experiment

We choose the optimizer as SGD, set the learning rate to 0.01, use the cross-entropy loss function, and cut the training set into 167 batches with 300 as the training set batch size, and the number of epochs is 20.The device we use is Nvidia Geforce RTX 3050 Laptop GPU.

In Fig 6,We plot the training loss, training accuracy, and testing accuracy during model training. In Table 2, we give the GPU time and final accuracy for a single epoch versus all (20) epochs.

4 GoogLeNet

4.1 The Architecture of GoogLeNet

In Fig 7, we give the model construction diagram of GoogLeNet.

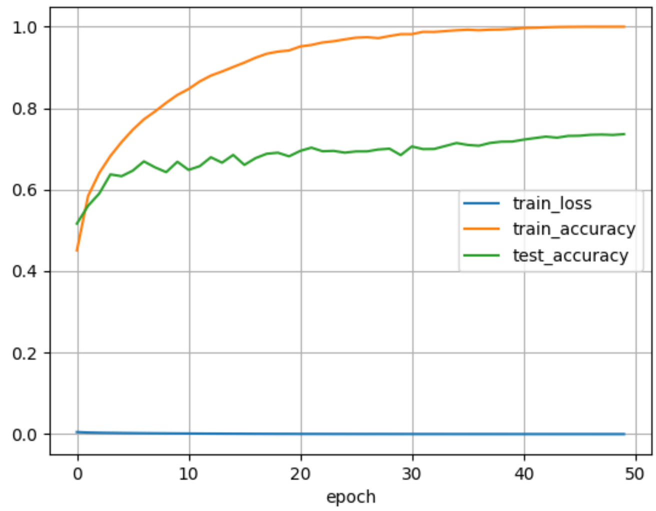
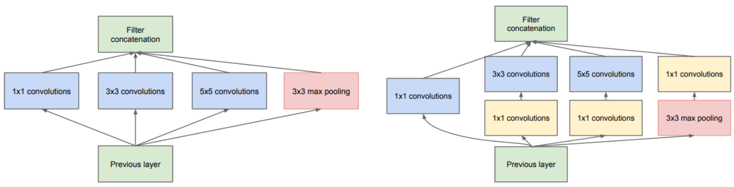


Figure 6: Training loss, training accuracy, and testing accuracy during model training

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Figure 7: The Architecture of GoogLeNet



(a) Inception module, naïve version

(b) Inception module with dimension reductions

Figure 8: The Architecture of Inception

4.2 Features of GoogLeNet

1. Put forward the concept of Inception.Fig 8 shows the structure of the Inception.