



✓ **Congratulations! You passed!**  
TO PASS 80% or higher

Keep Learning

GRADE  
100%

## Deep Convolutional Models

LATEST SUBMISSION GRADE

100%

1. Which of the following do you typically see in a ConvNet? (Check all that apply.)

1 / 1 point

☐ FC layers in the first few layers

☒ FC layers in the last few layers

✓ **Correct**

True, fully-connected layers are often used after flattening a volume to output a set of classes in classification.

☒ Multiple CONV layers followed by a POOL layer

✓ **Correct**

True, as seen in the case studies.

☐ Multiple POOL layers followed by a CONV layer

2. In order to be able to build very deep networks, we usually only use pooling layers to downsize the height/width of the activation volumes while convolutions are used with "valid" padding. Otherwise, we would downsize the input of the model too quickly.

1 / 1 point

☐ True

☒ False

✓ **Correct**

Correct!

3. Training a deeper network (for example, adding additional layers to the network) allows the network to fit more complex functions and thus almost always results in lower training error. For this question, assume we're referring to "plain" networks.

1 / 1 point

☐ True

☒ False

✓ **Correct**

Correct, Resnets are here to help us train very deep neural networks.

4. The following equation captures the computation in a ResNet block. What goes into the two blanks above?

1 / 1 point

$$a^{[l+2]} = g(W^{[l+2]}g(W^{[l+1]}a^{[l]} + b^{[l+1]}) + b^{[l+2]} + \text{_____}) + \text{_____}$$

- ☐ 0 and  $z^{[l+1]}$ , respectively
- ☐  $z^{[l]}$  and  $a^{[l]}$ , respectively
- ☐ 0 and  $a^{[l]}$ , respectively
- ☒  $a^{[l]}$  and 0, respectively

✓ **Correct**  
Correct

5. Which ones of the following statements on Residual Networks are true? (Check all that apply.)

1 / 1 point

- ☐ A ResNet with L layers would have on the order of  $L^2$  skip connections in total.
- ☒ The skip-connection makes it easy for the network to learn an identity mapping between the input and the output within the ResNet block.

✓ **Correct**  
This is true.

- ☐ The skip-connections compute a complex non-linear function of the input to pass to a deeper layer in the network.
- ☒ Using a skip-connection helps the gradient to backpropagate and thus helps you to train deeper networks

✓ **Correct**  
This is true.

6. Suppose you have an input volume of dimension  $n_H \times n_W \times n_C$ . Which of the following statements you agree with? (Assume that "1x1 convolutional layer" below always uses a stride of 1 and no padding.)

1 / 1 point

- ☐ You can use a 1x1 convolutional layer to reduce  $n_H$ ,  $n_W$ , and  $n_C$ .
- ☒ You can use a 1x1 convolutional layer to reduce  $n_C$  but not  $n_H$ ,  $n_W$ .

✓ **Correct**  
Yes, a 1x1 convolutional layer with a small number of filters is going to reduce  $n_C$  but will keep the dimensions  $n_H$  and  $n_W$

- ☒ You can use a 2D pooling layer to reduce  $n_H$ ,  $n_W$ , but not  $n_C$ .

✓ **Correct**

✓ **Correct**

This is correct.

- ☐ You can use a 2D pooling layer to reduce  $n_H$ ,  $n_W$ , and  $n_C$ .

7. Which ones of the following statements on Inception Networks are true? (Check all that apply.)

1 / 1 point

- ☐ Making an Inception network deeper (by stacking more Inception blocks together) *might* not hurt training set performance.
- ☒ A single Inception block allows the network to use a combination of 1x1, 3x3, 5x5 convolutions and pooling.

✓ **Correct**

- ☐ Inception networks incorporate a variety of network architectures (similar to dropout, which randomly chooses a network architecture on each step) and thus has a similar regularizing effect as dropout.
- ☒ Inception blocks usually use 1x1 convolutions to reduce the input data volume's size before applying 3x3 and 5x5 convolutions.

✓ **Correct**

8. Which of the following are common reasons for using open-source implementations of ConvNets (both the model and/or weights)? Check all that apply.

1 / 1 point

- ☒ It is a convenient way to get working with an implementation of a complex ConvNet architecture.

✓ **Correct**

True

- ☐ A model trained for one computer vision task can usually be used to perform data augmentation even for a different computer vision task.
- ☐ The same techniques for winning computer vision competitions, such as using multiple crops at test time, are widely used in practical deployments (or production system deployments) of ConvNets.
- ☒ Parameters trained for one computer vision task are often useful as pretraining for other computer vision tasks.

✓ **Correct**

True

9. In Depthwise Separable Convolution you:

1 / 1 point

- ☒ For the "Depthwise" computations each filter convolves with only one corresponding color channel of the input image.

✓ **Correct**

- ☐ Perform one step of convolution.
- ☐ You convolve the input image with a filter of  $n_f \times n_f \times n_c$  where  $n_c$  acts as the depth of the filter ( $n_c$  is the number of color channels of the input image).
- ☒ The final output is of the dimension  $n_{out} \times n_{out} \times n'_c$  (where  $n'_c$  is the number of filters used in the previous convolution step).

✓ Correct

- ☐ The final output is of the dimension  $n_{out} \times n_{out} \times n_c$  (where  $n_c$  is the number of color channels of the input image).
- ☒ Perform two steps of convolution.

✓ Correct

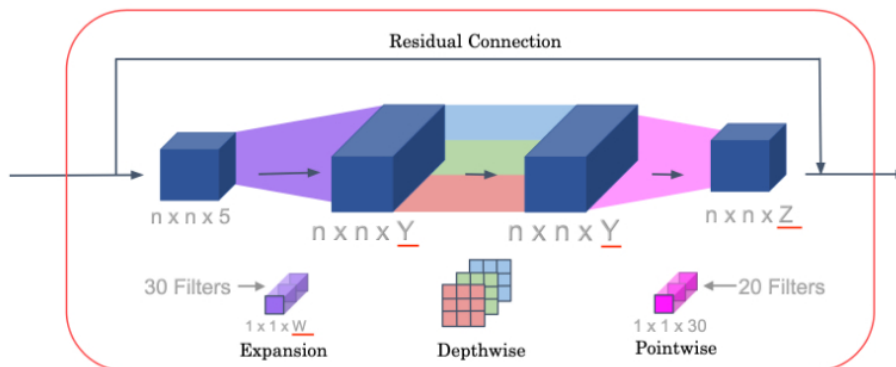
- ☐ For the "Depthwise" computations each filter convolves with all of the color channels of the input image.
- ☒ You convolve the input image with  $n_c$  number of  $n_f \times n_f$  filters ( $n_c$  is the number of color channels of the input image).

✓ Correct

10. Fill in the missing dimensions shown in the Image below (marked  $W$ ,  $Y$ ,  $Z$ ).

1 / 1 point

## MobileNet v2 Bottleneck



- ☐  $W = 5, Y = 20, Z = 5$
- ☒  $W = 5, Y = 30, Z = 20$
- ☐  $W = 30, Y = 20, Z = 20$
- ☐  $W = 30, Y = 30, Z = 5$

✓ Correct