Congratulations! You passed!

Grade received 80% **To pass** 80% or higher

Go to next item

The Basics of ConvNets

Latest Submission Grade 80%

1. What do you think applying this filter to a grayscale image will do? **1 / 1 point**

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 3 & 3 & 1 \\ -1 & -3 & -3 & -1 \\ 0 & -1 & -1 & 0 \end{bmatrix}$$

- (•) Detect horizontal edges.
- ( ) Detect 45-degree edges.
- ( ) Detecting image contrast.
- ( ) Detect vertical edges.

Expand

Correct
Correct. There is a high difference between the values in the top part from those in the bottom part of the matrix. When convolving this filter on a grayscale image, the horizontal edges will be detected.

2. Suppose your input is a 128 by 128 color (RGB) image, and you are not using a convolutional network. If the first hidden layer has 64 neurons, each one fully connected to the input, how many parameters does this hidden layer have (including the bias parameters)? **0 / 1 point**

- ( ) 1048640
- ( ) 1048576
- (•) 3145728
- ( ) 3145792

Expand

Incorrect
No, check that you add the bias parameters for each unit.

3. Suppose your input is a 256 by 256 grayscale image, and you use a convolutional layer with 128 filters that are each 3×3. How many parameters does this hidden layer have (including the bias parameters)? **1 / 1 point**

- ( ) 3584
- ( ) 75497600
- ( ) 1152
- (•) 1280

Expand

Correct
Yes, since the input volume has only one channel each filter has $3 \times 3 + 1$ weights including the bias, thus the total is $(3 \times 3 + 1) \times 128$.

4. You have an input volume that is 63x63x16, and convolve it with 32 filters that are each 7x7, using a stride of 2 and no padding. What is the output volume? **1 / 1 point**

- ( ) 16x16x32
- (•) 29x29x32
- ( ) 29x29x16
- ( ) 16x16x16

Expand

Correct
Yes, $\frac{63-7}{2}+1 = 29$ and the number of channels should match the number of filters.

5. You have an input volume that is 31x31x32, and pad it using "pad=1". What is the dimension of the resulting volume (after padding)? **1 / 1 point**

- (•) 33x33x32
- ( ) 31x31x34
- ( ) 32x32x32
- ( ) 33x33x33

Expand

Correct
Yes, if the padding is 1 you add 2 to the height dimension and 2 to the width dimension.

6. You have a volume that is $121 \times 121 \times 32$, and convolve it with 32 filters of 5×5, and a stride of 1. You want to use a "same" convolution. What is the padding? **1 / 1 point**

- (•) 2
- ( ) 5
- ( ) 0
- ( ) 3

Expand

Correct
Yes, when using a padding of 2 the output volume has $n_H = \frac{121+2\times2}{1}+1$.

7. You have an input volume that is 66x66x21, and apply max pooling with a stride of 3 and a filter size of 3. What is the output volume? **1 / 1 point**

- (•) $22 \times 22 \times 21$
- ( ) $66 \times 66 \times 7$
- ( ) $21 \times 21 \times 21$
- ( ) $22 \times 22 \times 7$

Expand

Correct
Yes, using the formula $n_H^{[l]} = \frac{n_H^{[l-1]} + 2 \times p - f}{s} + 1$ with $p = 0, f = 3, s = 3$ and $n_H^{[l-1]} = 66$.

8. Because pooling layers do not have parameters, they do not affect the backpropagation (derivatives) calculation. **1 / 1 point**

- ( ) True
- (•) False

Expand

Correct
Everything that influences the loss should appear in the backpropagation because we are computing derivatives. In fact, pooling layers modify the input by choosing one value out of several values in their input volume. Also, to compute derivatives for the layers that have parameters (Convolutions, Fully-Connected), we still need to backpropagate the gradient through the Pooling layers.

9. Which of the following are the benefits of using convolutional layers? (Check all that apply) **0 / 1 point**

- [ ] It reduces the computations in backpropagation since we omit the convolutional layers in the process.
- [x] It allows parameters learned for one task to be shared even for a different task (transfer learning).

! **This should not be selected**
No, transfer learning is not bound to ConvNets and can be used with other types of models as you've seen in Course 1-3.

- [x] It reduces the total number of parameters, thus reducing overfitting through parameter sharing.

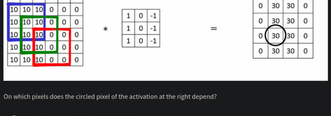
✓ **Correct**
Yes, a convolutional layer uses parameters sharing and has usually a lot fewer parameters than a fully-connected layer.

- [ ] Convolutional layers are good at capturing translation invariance.

Expand

Incorrect
You didn't select all the correct answers

10. The following image depicts the result of a convolution at the right when using a stride of 1 and the filter is shown right next. **1 / 1 point**



On which pixels does the circled pixel of the activation at the right depend?

- ( ) It depends on the pixels enclosed by the blue square.
- ( ) It depends on the pixels enclosed by the red square.
- ( ) It depends on all the pixels of the image on the left.
- (•) It depends on the pixels enclosed by the green square.

Expand

Correct
Yes, this is the position of the filter when we move it two pixels down and one to the right.