S2 - Quiz

Due 13 May at 9:00 **Points** 100 **Questions** 9
Available 6 May at 13:00 - 13 May at 9:00 **Time limit** 30 Minutes

Instructions

Instructions:

1. You have 30 minutes to attempt the quiz
2. Once you start the quiz, you cannot go back and re-attempt it
3. You will not find answers online, so please make sure you are ready for the quiz
4. For Multiple Answer Questions, ALL the answers must be correct to score any point

Sometimes you might see multiple empty options. Please do not consider those empty options, that's some rendering issue, the options you see are the only options available for that question.

Attempt history

	Attempt	Time	Score
LATEST	Attempt 1	5 minutes	97.5 out of 100

Score for this quiz: **97.5** out of 100
Submitted 12 May at 10:39
This attempt took 5 minutes.

Question 1

10 / 10 pts

If we perform convolution with a kernel of size 3x3 on 47x49, the output size would be?

○ Convolution cannot be performed on a rectangular channel

○ 45x45

Correct! ● 45x47

○ 47x47

Question 2

7.5 / 10 pts

Which of these are true, w.r.t. what we discussed in Session

Correct! ☑ We add as many layers as required to reach full image/object size

orrect answer ☐ We normally add padding to keep the output channel size same as the input channel

Correct! ☑ We nearly always use kernels with stride of 1, unless we need to pool the layers

Correct! ☑ We always use a kernel with size 3x3

Question 3

10 / 10 pts

How many 3x3 layers do we need to add to reach a receptive field of 21x21?

○ 11

○ 9

○ 12

Correct! ● 10

Question 4

10 / 10 pts

Let us assume we have an image of size 100x100. What is the minimum number of **convolution layers** do we need to add such that

1. you cannot use max-pooling without convolving twice or more
2. the output is at least 2-3 convolution layers away from max-pooling
3. You can stop either at 2x2 or 1x1 based on how you have used your layers
4. we will always "not consider" the last rows and columns in an odd-resolution channel while performing max-pooling)
5. "do not" count the max-pooling layer

Correct! ● 10

○ 9

○ 11

○ 13

Question 5

10 / 10 pts

If the input layer has 128 channels, how many kernels do we need to add?

○ Exactly 128

Correct! ● Number of Kernels do not depend on input channels

○ 32

○ 64

Question 6

10 / 10 pts

Consider the following layers

...

49x49x256 | Convolved with 512 kernels of size 3x3 |

...

What is the total number of kernel parameters we just added?

Correct!

● 1179648

○ 2304

○ 4608

○ 314703872

Question 7

10 / 10 pts

Consider this network

400x400x3 | 32x(3x3x3) |
398x398x32 | 64x(3x3x32) |
396x396x64 | 128x(3x3x64) |
394x394x128 | 256x(3x3x128) |
392x392x256 | 512x(3x3x256) |
390x390x512 | 1024x(3x3x512) |
MaxPooling(2x2)
...

Assume this network is trained and we are doing inference on an image. Before we hit the max-pooling layer, how many channels of size more than 350x350 are there in the GPU RAM?

○ None of them are correct 😡🤬😡

○ 2016

Correct! ● 2019

○ 992

○ 995

Question 8

15 / 15 pts

What are the few advantages of using MaxPooling?

Correct! ☑ Reduction in Channel Size

Correct! ☑ Slight Rotational Invariance

Correct! ☑ Slight Translational Invariance

□ Reduction in Number of Channels

Question 9

15 / 15 pts

If we start with an image of 400x400 color, and during a model we use MaxPooling 4 times, reducing the image size to 400>200>100>50 (we used convs with padding, so convs did not reduce the image size), have we lost 4 times the information we started with? At 50x50 we have 1000 channels.

□ Yes, that's correct, that is what information theory would predict

Correct!

☑

No, convs and poolings operation are loosing some information, but more importantly, they are "filtering" the information. We do not need full information at the last layer, just the most important one. We are also scaling in Z axis (from 3 to 1000), and it is the increase in z axis where we store this "proposed" lost information.

□

No, that is incorrect. Since images are 2D we actually had 400x400 units of information, and we ended with 50x50. So total loss is 400x400/50/50 = 64 times

□

No, that is incorrect. Since image is actually 400x400x3, and we ended at 50x50x1000, we have lost 400x400x3/50/50/1000 = 0.192. So we have actually gained around 5 times more information

Quiz score: **97.5** out of 100