MONASH University

FIT5215 Deep Learning

Quiz for: Convolutional Neural Network

Trung Le and Tutor Team

Department of Data Science and AI
Faculty of Information Technology, Monash University
Email: **trunglm@monash.edu**

GROUP OF EIGHT AUSTRALIA

Question 1

*Given an 3D input tensor with shape [3, 32, 32] over which we apply a conv2D with **16 filters** each of which has shape **[3,5,5]**, strides **[3,3]**, and padding=**0**. What is the shape of the output tensor?*

- ❑ A. [10, 10, 16]
- ❑ B. [11, 11, 16]
- ❑ C. [16 , 11, 11]
- ❑ D. [16 , 10, 10]

Question 1

*Given an 3D input tensor with shape [3, 32, 32] over which we apply a conv2D with **16 filters** each of which has shape **[3,5,5]**, strides **[3,3]**, and padding=**0**. What is the shape of the output tensor?*

$$\left\lfloor \frac{32 + 2 \times 0 - 5}{3} \right\rfloor + 1 = 10$$

- ❑A. [10, 10, 16]
- ❑B. [11, 11, 16]
- ❑C. [16, 11, 11]
- ❑D. [16 ,10, 10] **[x]**

Question 2

*Given an 3D input tensor with shape [3, 32, 32] over which we apply a conv2D with **filters** which has shape **[32, 3, 5, 5]**, strides **[3,3]**, and padding=**2**. What is the shape of the output tensor?*

- ❑A. [32,10,10]
- ❑B. [32,11, 11]
- ❑C. [11,11,16]
- ❑D. [10,10,16]

Question 2

Given an 3D input tensor with shape [3, 32, 32] over which we apply a conv2D with ***filters*** *which has shape* ***[32, 3, 5, 5]****, strides* ***[3,3]****, and padding=****2****. What is the shape of the output tensor?*

$$\left\lfloor \frac{32 + 2 \times 2 - 5}{3} \right\rfloor + 1 = 11$$

- ❑A. [32,10,10]
- ❑B. [32,11, 11] [**x**]
- ❑C. [11,11,16]
- ❑D. [10,10,16]

Question 3

*Given an 3D input tensor with shape [10, 64, 64] over which we apply a **max pooling** layer with kernel size **[3,3]**, strides [**2,2]**, and padding=1. What is the **shape** of the **output tensor**?*

- ❑A. [10, 31, 31]
- ❑B. [10, 32, 32]
- ❑C. [22, 22, 10]
- ❑D. [32, 32, 10]

Question 3

Given an 3D input tensor with shape [10, 64, 64] over which we apply a ***max pooling*** *layer with kernel size* ***[3,3]****, strides [***2,2***], and padding=1. What is the* ***shape*** *of the* ***output tensor****?*

- A. [10, 31, 31]
- B. [10, 32, 32] **[x]**
- C. [22, 22, 10]
- D. [32, 32, 10]

$$\left\lfloor \frac{64 + 2 \times 1 - 3}{2} \right\rfloor + 1 = 32$$

Question 4

*Assume that the tensor before the last tensor of a CNN has shape [32, 10, 32, 32] and we apply **20 filters** each of which has the shape **[10, 5, 5]** and strides= **[2,2]** with padding = **1** to obtain the last tensor. What is the **shape** of the output tensor?*

- ❑A. [20, 15, 15]
- ❑B. [20, 16, 16]
- ❑C. [32, 20, 16, 16]
- ❑D. [32, 20, 15, 15]

Question 4

*Assume that the tensor before the last tensor of a CNN has shape [32, 10, 32, 32] and we apply **20 filters** each of which has the shape **[10, 5, 5]** and strides= **[2,2]** with padding = **1** to obtain the last tensor. What is the **shape** of the output tensor?*

- ❑A. [20, 15, 15]
- ❑B. [20, 16, 16]
- ❑C. [32, 20, 16, 16]
- ❑D. [32, 20, 15, 15] **[x]**

- Given an image in a minibatch, we convolve each [10 , 32, 32] with [10 , 5, 5] to achieve a 15x15 feature map.

$$\left\lfloor \frac{32 + 2 \times 1 - 5}{2} \right\rfloor + 1 = 15$$

- There are **20** filters → [20, 15, 15]
- There are **32** inputs in a minibatch → [**32**, 20, 15, 15]

Question 5

Assume that the tensor before the last tensor of a CNN has shape [64, 10, 32, 32] and we apply ***5 filters*** *each of which has the shape* ***[10, 5,5]*** *and strides=* ***[2,2]*** *with padding =* ***0*** *to obtain the last tensor. We flatten this tensor to a* ***fully connected*** *(FC) layer. What is the* ***number of neurons*** *on this FC layer?*

- ❑A. 14 x 14
- ❑B. 14 x 14 x 5
- ❑C. 64 x 16 x 16 x5
- ❑D. 64 x 14 x 14 x 5

Question 5

*Assume that the tensor before the last tensor of a CNN has shape [64, 10, 32, 32] and we apply **5 filters** each of which has the shape **[10, 5,5]** and strides= **[2,2]** with padding = **0** to obtain the last tensor. We flatten this tensor to a **fully connected** (FC) layer. What is the **number of neurons** on this FC layer?*

- ❑A. 14 x 14
- ❑B. 14 x 14 x 5 **[x]**
- ❑C. 64 x 16 x 16 x5
- ❑D. 64 x 14 x 14 x 5

- Given an image in a minibatch, we convolve each [10 , 32, 32] with [10, 5, 5] to achieve a 14x14 feature map.

$$\left\lfloor \frac{32 + 2 \times 0 - 5}{2} \right\rfloor + 1 = 14$$

- There are **5** filters → [**5 ,** 14, 14]
- We flatten [5 , 14, 14] and obtain 14x14x5 neurons

Question 6

What likely happen if using a large filter (e.g., 7x7, 9x9) with a deep model (e.g., 20 layers) if there are few images?

- ❑A. Overfitting
- ❑B. Underfitting

Question 6

What likely happen if using a large filter (e.g., 7x7, 9x9) with a deep model (e.g., 20 layers) if there are few images?

- ❑ A. Overfitting **[x]**
- ❑ B. Underfitting

- Larger filter → More parameters → Overfitting problem
- Small filter → Fewer parameters → Underfitting problem
- If we use large enough model (several layers), 3x3 filter will be a common choice.

Question 7

Which is a good CNN model architecture?

- ❑A. Input layer → Convolutional layer (Activation) → Pooling layer → FC layer → Output
- ❑B. Input layer → Pooling layer → Convolutional layer (Activation) → FC layer → Output
- ❑C. Input layer → FC Layer → Pooling layer → Convolutional layer (Activation) → Output
- ❑D. Input layer → Convolutional layer (Activation) → FC layer → Pooling layer → Output

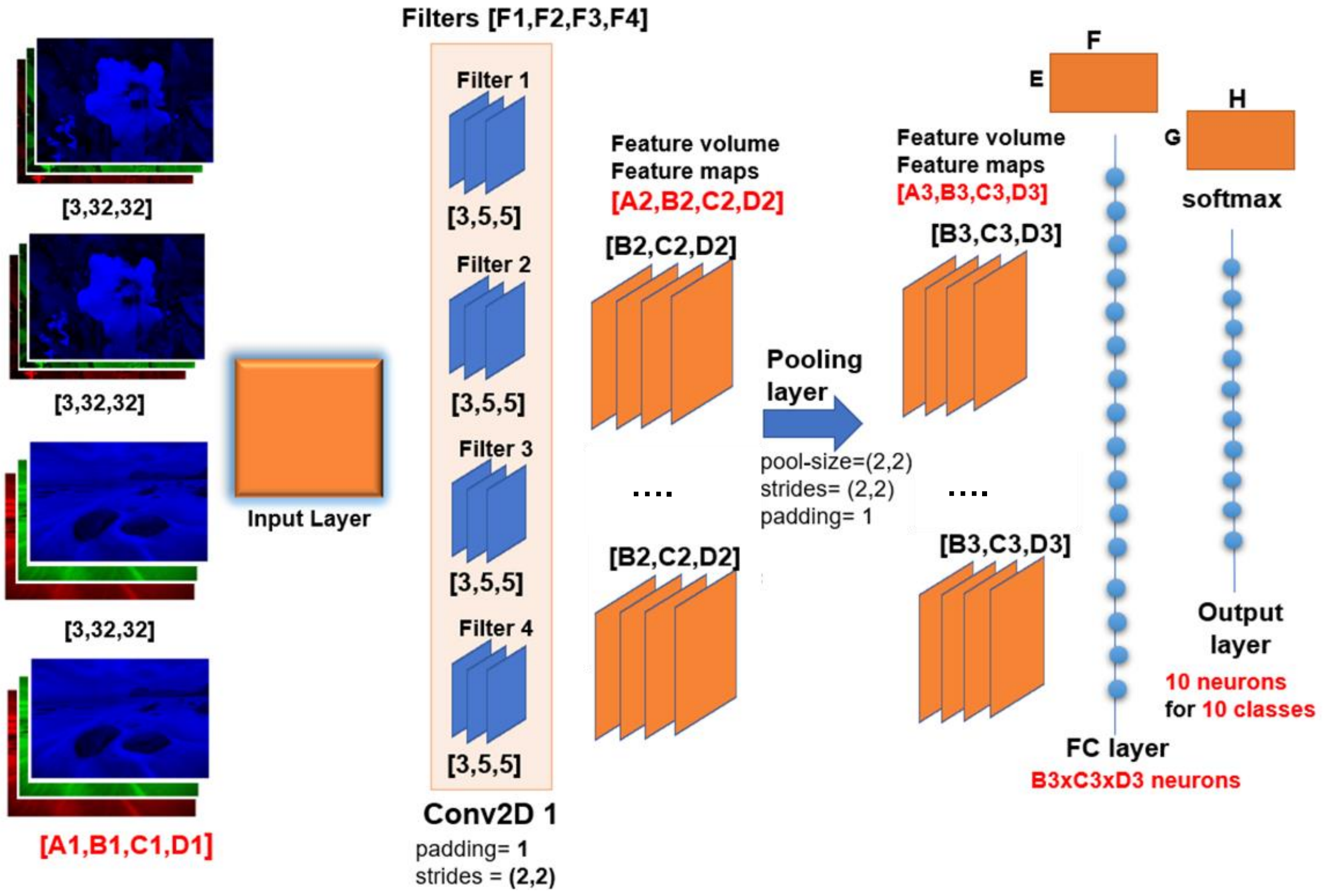
Question 7

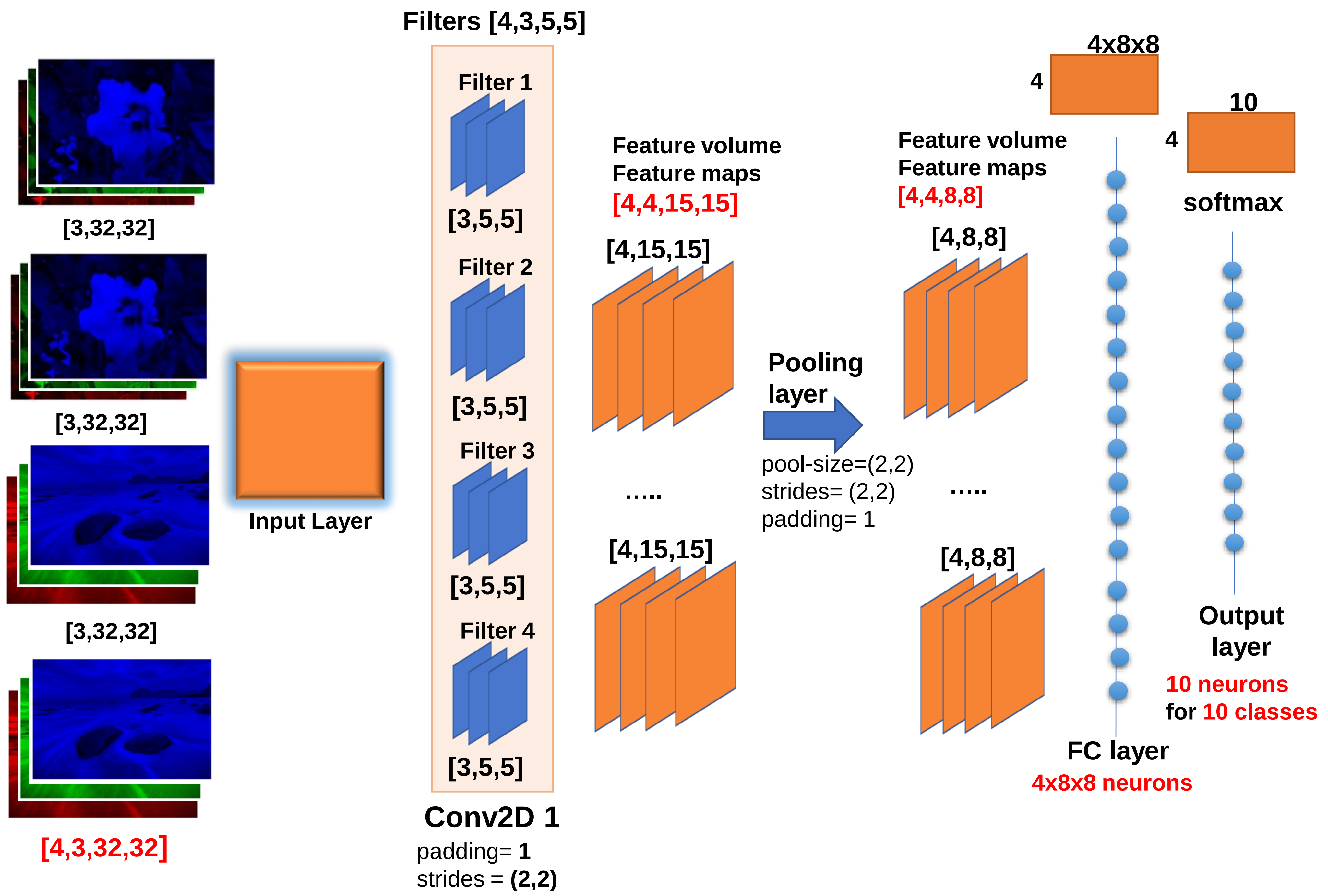
Which is a good CNN model architecture?

- ❑A. Input layer → Convolutional layer (Activation) → Pooling layer → FC layer → Output **[x]**
- ❑B. Input layer → Pooling layer → Convolutional layer (Activation) → FC layer → Output
- ❑C. Input layer → FC Layer → Pooling layer → Convolutional layer (Activation) → Output
- ❑D. Input layer → Convolutional layer (Activation) → FC layer → Pooling layer → Output

Question 8

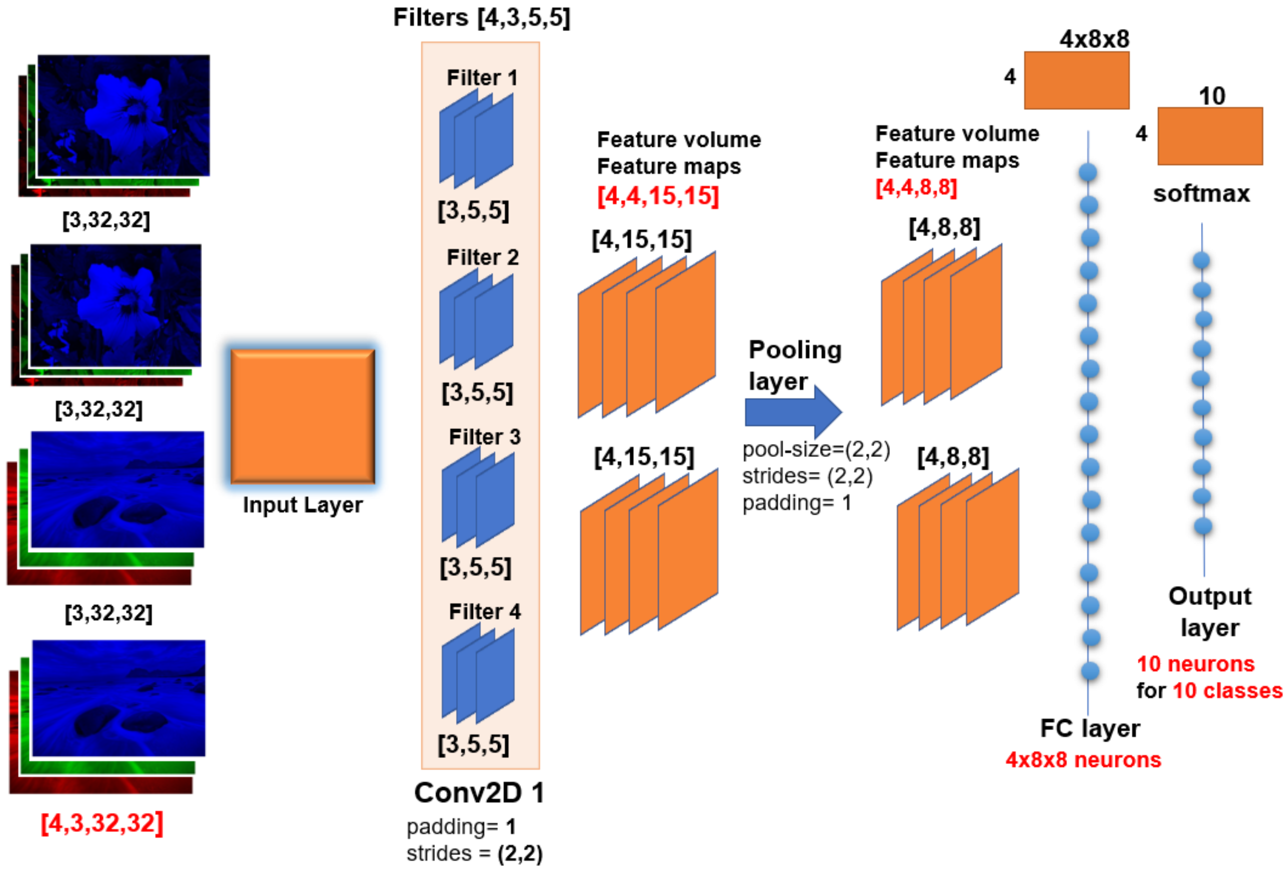
- *Consider the following CNN in operation. What are* ***[A1,B1,C1,D1]****,* ***[F1, F2, F3, F4]****,* ***[A2,B2,C2,D2]****,****[A3,B3,C3, D3]****,* ***[E,F]****,* ***[G,H]****?*




Filters [4,3,5,5]
Filter 1
[3,5,5]
Filter 2
[3,5,5]
Filter 3
[3,5,5]
Filter 4
[3,5,5]
Conv2D 1
padding= 1
strides = (2,2)
[3,32,32]
[3,32,32]
[3,32,32]
[4,3,32,32]
Input Layer
Feature volume
Feature maps
[4,4,15,15]
[4,15,15]
.....
[4,15,15]
Pooling layer
pool-size=(2,2)
strides= (2,2)
padding= 1
Feature volume
Feature maps
[4,4,8,8]
[4,8,8]
.....
[4,8,8]
4x8x8
4
10
4
softmax
FC layer
4x8x8 neurons
Output layer
10 neurons
for 10 classes

Question 9

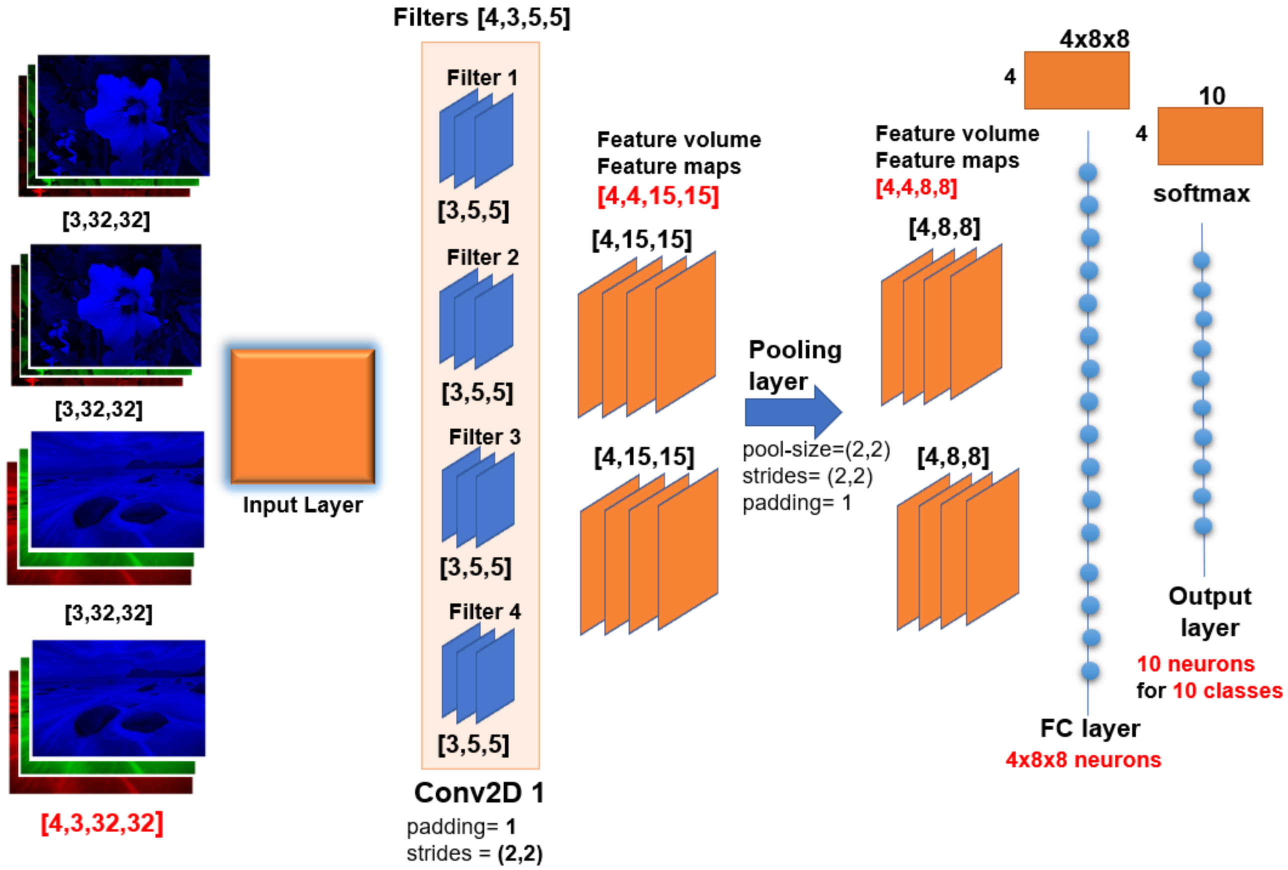
*Consider the following CNN in operation. Assume that the 2D tensor [4,10] at the output layer is the one **after applied softmax**. What is the meaning of the **second row** in this tensor?*



- ❑A. The prediction probabilities of the second image in the batch.
- ❑B. The logits of the second image in the batch.
- ❑C. The prediction probabilities of the third image in the batch.
- ❑D. The logits of the third image in the batch.

Question 9

Consider the following CNN in operation. Assume that the 2D tensor [4,10] at the output layer is the one ***after applied softmax****. What is the meaning of the* ***second row*** *in this tensor?*



- ❑A. The prediction probabilities of the second image in the batch. **[x]**
- ❑B. The logits of the second image in the batch.
- ❑C. The prediction probabilities of the third image in the batch.
- ❑D. The logits of the third image in the batch.

Question 10

What is the purpose of the Conv2D layers? (MC)

- ❑A. Process 1D tensors.
- ❑B. Process 2D and 3D tensors.
- ❑C. Provide some filters to detect the patterns inside an input tensor.
- ❑D. Reduce the input size to make it more manageable.

Question 10

What is the purpose of the Conv2D layers?

- ❑A. Process 1D tensors.
- ❑B. Process 2D and 3D tensors. **[x]**
- ❑C. Provide some filters to detect the patterns inside an input tensor. **[x]**
- ❑D. Reduce the input size to make it more manageable.

Question 11

What are correct about the pooling layers? (MC)

- ❑A. Preserve the crucial information in the input.
- ❑B. Increase the input size to have more information.
- ❑C. Provide some filters to detect the patterns inside an input tensor.
- ❑D. Reduce the input size to make it more manageable.

Question 11

What are correct about the pooling layers? (MC)

- ❑A. Preserve the crucial information in the input. **[x]**
- ❑B. Increase the input size to have more information.
- ❑C. Provide some filters to detect the patterns inside an input tensor.
- ❑D. Reduce the input size to make it more manageable. **[x]**

Question 12

What are correct about the BatchNorm layers? (MC)

- ❑A. Mitigate the overfitting.
- ❑B. In training phase, it uses the population statistics (mean and std).
- ❑C. In training phase, it uses the batch statistics (mean and std).
- ❑D. In testing phase, it uses the population statistics (mean and std).
- ❑E. In testing phase, it uses the batch statistics (mean and std)

Question 12

What are correct about the BatchNorm layers? (MC)

- ❑A. Mitigate the overfitting. **[x]**
- ❑B. In training phase, it uses the population statistics (mean and std).
- ❑C. In training phase, it uses the batch statistics (mean and std). **[x]**
- ❑D. In testing phase, it uses the population statistics (mean and std). **[x]**
- ❑E. In testing phase, it uses the batch statistics (mean and std)

Question 13

Given an implementation as below. What is the shape printed?

```
cnn_part = nn.Sequential(
    nn.Conv2d(3, 32, kernel_size=3, stride= 2,padding=1),
    nn.ReLU(),
    nn.Conv2d(32, 64, kernel_size=3, stride =1, padding=1),
    nn.ReLU()
)

x= torch.randn(1,3,224,224)
print(cnn_part(x).shape)
```

- ❑ A. [1, 64, 112, 112]
- ❑ B. [32, 64, 112, 112]
- ❑ C. [32, 64, 224, 224]
- ❑ D. [1, 64, 224, 224]

Question 13

Given an implementation as below. What is the shape printed?

```
cnn_part = nn.Sequential(
     nn.Conv2d(3, 32, kernel_size=3, stride= 2,padding=1),
     nn.ReLU(),
     nn.Conv2d(32, 64, kernel_size=3, stride =1, padding=1),
     nn.ReLU()
)

x= torch.randn(1,3,224,224)
print(cnn_part(x).shape)
```

- ❑ A. [1, 64, 112, 112] **[x]**
- ❑ B. [32, 64, 112, 112]
- ❑ C. [32, 64, 224, 224]
- ❑ D. [1, 64, 224, 224]

Question 14

Given an implementation as below. What is the shape printed?

```
cnn_part = nn.Sequential(
    nn.Conv2d(3, 32, kernel_size=3, stride= 2,padding=1),
    nn.ReLU(),
    nn.Conv2d(32, 64, kernel_size=3, stride =1, padding=1),
    nn.BatchNorm2d(64),
    nn.ReLU(),
    nn.MaxPool2d(kernel_size=2, stride=2, padding = 0)
)

x= torch.randn(64,3,224,224)
print(cnn_part(x).shape)
```

- A. [1, 64, 112, 112]
- B. [64, 64, 56, 56]
- C. [64, 64, 224, 224]
- D. [1, 64, 56, 56]

Question 14

Given an implementation as below. What is the shape printed?

```
cnn_part = nn.Sequential(
    nn.Conv2d(3, 32, kernel_size=3, stride= 2,padding=1),
    nn.ReLU(),
    nn.Conv2d(32, 64, kernel_size=3, stride =1, padding=1),
    nn.BatchNorm2d(64),
    nn.ReLU(),
    nn.MaxPool2d(kernel_size=2, stride=2, padding = 0)
)

x= torch.randn(64,3,224,224)
print(cnn_part(x).shape)
```

- A. [1, 64, 112, 112]
- B. [64, 64, 56, 56] **[x]**
- C. [64, 64, 224, 224]
- D. [1, 64, 56, 56]

Question 15

Given an implementation as below. What is the shape printed?

```
cnn_part = nn.Sequential(
    nn.Conv2d(3, 32, kernel_size=3, stride= 2,padding=1),
    nn.ReLU(),
    nn.Conv2d(32, 64, kernel_size=3, stride =1, padding=1),
    nn.BatchNorm2d(64),
    nn.ReLU(),
    nn.MaxPool2d(kernel_size=2, stride=2, padding = 0),
    nn.Flatten(1),
    nn.Linear(64*56*56, 26)
)

x= torch.randn(64,3,224,224)
print(cnn_part(x).shape)
```

- A. [64, 64*56*56]
- B. [64, 26]
- C. [64, 64, 56, 56]
- D. [1, 26]

Question 15

Given an implementation as below. What is the shape printed?

```
cnn_part = nn.Sequential(
    nn.Conv2d(3, 32, kernel_size=3, stride= 2,padding=1),
    nn.ReLU(),
    nn.Conv2d(32, 64, kernel_size=3, stride =1, padding=1),
    nn.BatchNorm2d(64),
    nn.ReLU(),
    nn.MaxPool2d(kernel_size=2, stride=2, padding = 0),
    nn.Flatten(1),
    nn.Linear(64*56*56, 26)
)

x= torch.randn(64,3,224,224)
print(cnn_part(x).shape)
```

- ❑ A. [64, 64*56*56]
- ❑ B. [64, 26] **[x]**
- ❑ C. [64, 64, 56, 56]
- ❑ D. [1, 26]