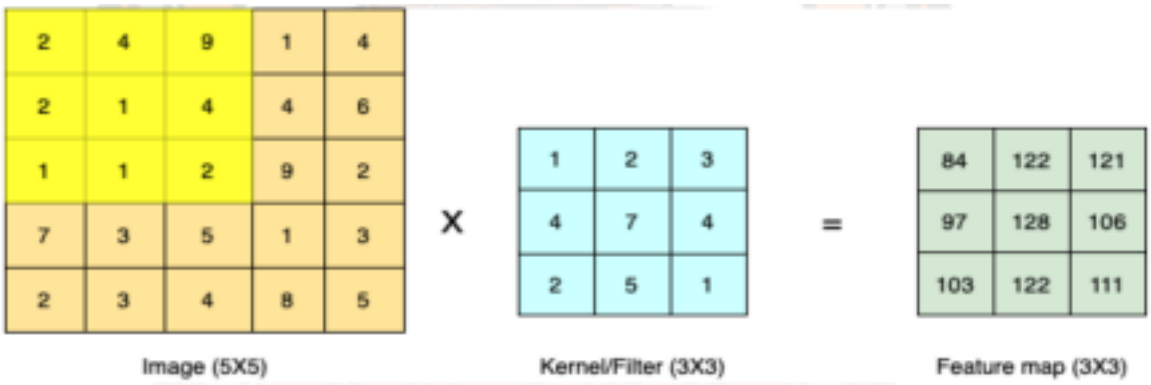
Exercise ABC: Submission Required!

Use the necessary formulae for the calculations below.

1. What will be the feature map dimension for the following:



Stride = 1

Answer: since the stride is 1 the feature map dimension will be 3 itself

2. You are going to perform the "same" padding to the above input image. For that, calculate how many pixels you need to add to the input image.

Answer

Given:

image_size=5x5

filter_size=3x3

Same_padding=f-1/2

=3-1/2

=2/2

=1

Therefore we need to add 1 extra pixel to the input image

3. Calculate the dimension of the newly padded image.

Answer:

Dimension of the newly padded image

Given:

w_input=5

h_input=5

padding=1

filter=3

stride=1

w_output=((w_input+2xp-f)/s)+1

=((5+2 x 1-3)/1)+1

w_output=5

h_output=((h_input+2xp-f)/s)+1

=((5+2 x 1-3)/1)+1

h_output=5

Therefore the dimensions of the newly padded image is (5,5)

4. Take the following values:

Input = 5x5

kernel size= 3x3

Stride = 1,

Padding = use the value retrieved in question (2)

Calculate the output shape for the feature map.

Answer:

w_output=((w_input+2xp-f)/s)+1

=((5+2 x 1-3)/1)+1

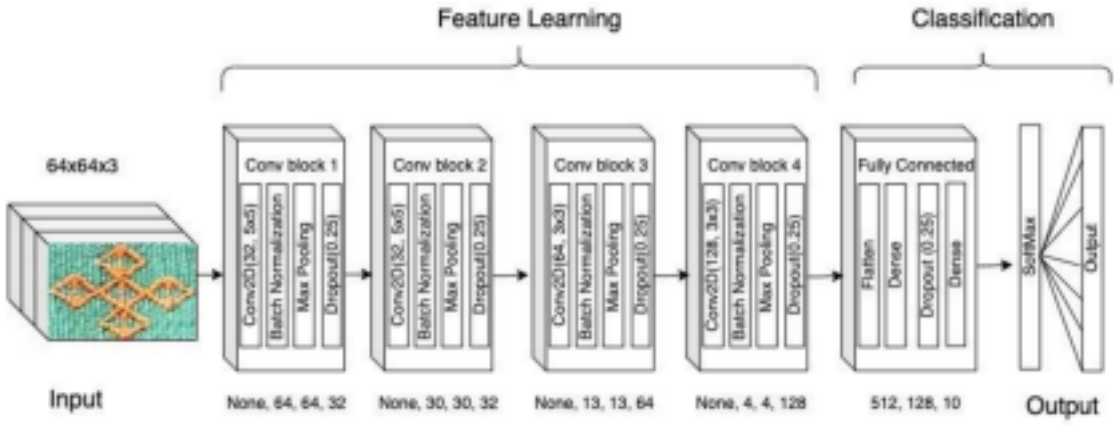
w_output=5

h_output=((h_input+2xp-f)/s)+1
=((5+2 x 1-3)/1)+1
h_output=5

Therefore the output shape for the feature map is (5,5)

5. Refer to the Convolutional Neural Network and its summary given below. Calculate the parameters in each CNN layer and fill the Parameter column in the table. Also, calculate the total number of parameters.

CNN Architecture:



Model Summary:

Calculation of Parameters for Conv2D Layers

param_number = output_channel_number * (input_channel_number * kernel_height * kernel_width + 1)

Calculation of Parameters for BatchNormalisation

Parameters=2×Number of Channels

Calculation of Parameters for Dense

Parameters=(Number of Input Nodes+1)×Number of Output Nodes

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 64, 64, 32)	Parameters=(5×5×1+1)×32=832

	kernel size: 5x5	
batch_normalization (BatchNormalization)	(None, 64, 64, 32)	2 x 32 = 64
max_pooling2d (MaxPooling2D)	(None, 32, 32, 32)	0
conv2d_1 (Conv2D)	(None, 32, 32, 32) kernel size: 5x5	Parameters=(5×5×32+1)×32=256 32
batch_normalization_1 (BatchNormalization)	(None, 32, 32, 32)	2*32=64
max_pooling2d_1 (MaxPooling 2D)	(None, 16, 16, 32)	0
conv2d_2 (Conv2D)	(None, 16, 16, 64) kernel size: 3x3	Parameters=(3×3×32+1)×64=18 496
batch_normalization_2 (BatchNormalization)	(None, 16, 16, 64)	2 x 64 = 128
max_pooling2d_2 (MaxPooling 2D)	(None, 8, 8, 64)	0
conv2d_3 (Conv2D)	(None, 8, 8, 64)	Parameters=(3×3×64+1)×64=36 928
batch_normalization_3 (BatchNormalization)	(None, 8, 8, 64)	2 x 64 = 128
max_pooling2d_3 (MaxPooling 2D)	(None, 4, 4, 64)	0
flatten (Flatten) (None, 1024)	(None, 1024)	0
dense (Dense) (None, 128)	(None, 128)	Parameters =(1024+1)×128=131,200
dense_1 (Dense) (None, 10)	(None, 10)	Parameters =(128+1)×10=1290

Total params:		**214,662.**