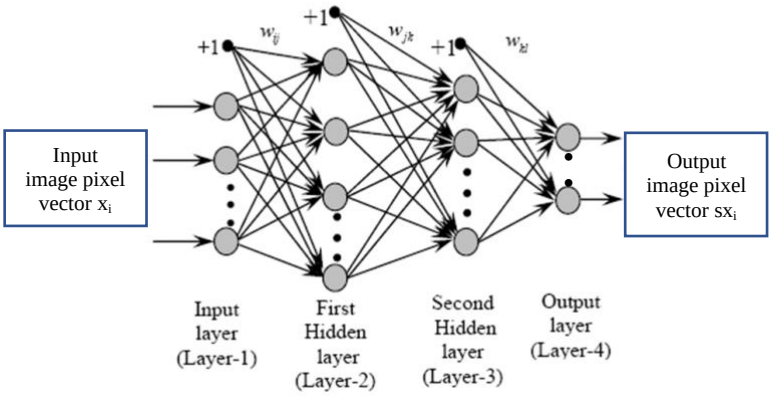
COMP4432 Machine Learning

Tutorial Questions on Convolutional Neural Networks (with answers)

1. Super resolution is the process of upscaling and/or improving the details within an image.

a) Suppose you are asked to use a multilayer perceptron (MLP) neural network as shown in Fig.1 below to learn a mapping from a given lower resolution grayscale (i.e. no color) image x_i to an upscaled grayscale image sx_i.



Assume that the input image size is 20×20 pixels and the upscaled image size is 50×50 pixels.

(i) How many learnable parameters, i.e. interconnecting weights, will be involved if there are N_{L2} hidden neurons in Layer-2 and N_{L3} hidden neurons in Layer-3? Note that there exist bias weights with fixed input +1 as shown in the MLP above.

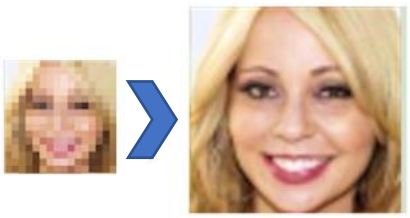
(ii) How training data should be collected and used in the MLP based super resolution model?

Answers of part (a):

(i) Size of input image pixel vector x_i: 20×20=400
Hence, the number of input-to-layer2 learnable parameters is $N_{12}=(401) \times N_{L2}$
The number of layer2-to-layer3 learnable parameters is $N_{23}=(N_{L2}+1) \times N_{L3}$
The number of layer3-to-output learnable parameters is $N_{34}=(N_{L3}+1) \times (50 \times 50)$
So, the total number of learnable parameters is: $N_{12}+N_{23}+N_{34}$

(ii) There may involve many aspects. A key issue is how to collect the training image pairs (20x20 input image; 50x50 output image). Hence, we need to collect sufficient number of images with resolution higher than 50x50, downscale them to 20x20 so that sufficient pairs of (20x20 input image, 50x50 output image) are prepared and can be used to feed to MLP for proper training. It is expected that this issue should be addressed. Other issues like how to deal with collected images with different resolution can also be discussed.

b) Suppose now the MLP in part (a) is enhanced with convolutional layers and pooling layers so that a CNN is resulted to carry out **color** image super resolution. Assume that 3 (color) channels are used to represent a color image.



(i) Show the convolution results of the following 6x6 image plane (1 plane only) with the associated 2 3x3 filters using stride=1. Here, no zero padding to the input image is applied.

1	0	1	0	0	2
0	3	0	0	1	0
1	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	2	0
0	0	1	0	3	0

6x6 image plane

1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

-1	1	-1
-1	1	-1
-1	1	-1

Filter 2

The convolution results (feature maps):

Filter 1

3	-4	-2	-2
-6	3	0	-3
-2	-3	1	0
3	-2	-1	-2

Filter 2

-1	-2	-2	-2
0	-3	-2	1
-2	-1	-3	2
-1	0	-7	6

(ii) For the feature maps generated from part (b-i), apply a 2x2 max pooling and show the result. Recall that a 2x2 max pooling is to select the maximum value from a group of 2x2 windowed values. Here, stride=2 is assumed.

The max pooling results:

Filter 1

3	0
3	1

Filter 2

0	1
0	6

(iii) For the following table of CNN architecture, how many learnable parameters are there in each of the specified layers? Show the formula or calculations in your answers.

Layer in CNN	Specification	Number of learnable parameters (formula answer is acceptable)
Input Layer	20x20 color images (3 channels)	0
1st Convolutional Layer	16 3x3x3 filters; stride=1; no zero padding	(3x3x3)x16
1st Max Pooling Layer	2x2 window; stride=2	0
2nd Convolutional Layer	64 3x3x16 filters; stride=1; no zero padding	(3x3x16)x64
Input layer of fully connected (fc) feedforward network	Just the flattened output from previous layer	0
1st hidden layer of fc feedforward network	N_{L2} hidden neurons	(7x7x64+1)xN_{L2}
2nd hidden layer of fc feedforward network	N_{L3} hidden neurons	(N_{L2}+1)xN_{L3}
Output layer	50x50 color images (3 channels)	(N_{L3}+1)x2500x3

There could also have bias term in the convolutional layers and the 1st ConvLayer will have (3x3x3+1)x16 learnable parameters.

Note here that the required answers are referring to the learnable parameters, rather than the processed results like feature maps, max pooling outputs, etc. For example, no matter how large the input image is, the number of learnable parameters in the first convolutional layer is still 16x3x3x3. The flattened number of inputs to fc layers is 7x7x64 because

Layer in CNN	Specification	Number of learnable parameters	Memory size
Input Layer	20x20 color images (3 channels)	0	20x20x3
1st Convolutional Layer	16 3x3x3 filters; stride=1; no zero padding	(3x3x3)x16	18x18x16
1st Max Pooling Layer	2x2 window	0	9x9x16
2nd Convolutional Layer	64 3x3x16 filters; stride=1; no zero padding	(3x3x16)x64	7x7x64
Input layer of fully connected (fc) feedforward network	Just the flattened output from previous layer	0	7x7x64
1st hidden layer of fc feedforward network	N_{L2} hidden neurons	(7x7x64+1)xN_{L2}	N_{L2}
2nd hidden layer of fc feedforward network	N_{L3} hidden neurons	(N_{L2}+1)xN_{L3}	N_{L3}
Output layer	50x50 color images (3 channels)	(N_{L3}+1)x2500x3	50x50x3