CS577 - Deep Learning:

Assignment 4:

Theoretical Questions

i Given:

I be a 4x4 RGB image.

R channel = $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ G = $\begin{bmatrix} 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}$ B = $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix}$

filter = $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

Convoluted image:

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}}_{R} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 9 & 9 \\ 9 & 9 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 18 & 18 \\ 18 & 18 \end{bmatrix}$$

$$\underset{B}{\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix}} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 18 & 18 \\ 27 & 27 \end{bmatrix}$$

Final Image: (Adding R, G, B channel)

$$\begin{bmatrix} 45 & 45 \\ 54 & 54 \end{bmatrix}$$

2 Prev. Question with zero padding.

$$\underset{R}{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 6 & 6 & 4 \\ 6 & 9 & 9 & 6 \\ 6 & 9 & 9 & 6 \\ 4 & 6 & 6 & 4 \end{bmatrix}$$

$$\underset{G}{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 2 & 2 & 2 & 0 \\ 0 & 2 & 2 & 2 & 2 & 0 \\ 0 & 2 & 2 & 2 & 2 & 0 \\ 0 & 2 & 2 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 8 & 12 & 12 & 8 \\ 12 & 18 & 18 & 12 \\ 12 & 18 & 18 & 12 \\ 8 & 12 & 12 & 8 \end{bmatrix}$$

Scanned with CamScanner

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 2 & 2 & 2 & 2 & 0 \\ 0 & 3 & 3 & 3 & 3 & 0 \\ 0 & 4 & 4 & 4 & 4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 6 & 9 & 9 & 6 \\ 12 & 18 & 18 & 12 \\ 18 & 27 & 27 & 18 \\ 14 & 21 & 21 & 14 \end{bmatrix}$$

B

Final Image (by adding R, G, B channel).

$$\begin{bmatrix} 18 & 27 & 27 & 18 \\ 30 & 45 & 45 & 30 \\ 36 & 54 & 54 & 36 \\ 26 & 39 & 39 & 26 \end{bmatrix}.$$

3. Prev. question with dilated conv. dilation rate = 2.

Having a dilation rate = 2 for a 3x3 filter results in a 5x5 filter with 0's in the even numbered rows and columns.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \longrightarrow \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

On convoluting with a 4x4 image that is zero padded.

$$\begin{bmatrix} 0&0&0&0&0&0\\ 0&1&1&1&1&0\\ 0&1&1&1&1&0\\ 0&1&1&1&1&0\\ 0&1&1&1&1&0\\ 0&0&0&0&0&0 \end{bmatrix} * \begin{bmatrix} 1&0&1&0&1\\ 0&0&0&0&0\\ 1&0&1&0&1\\ 0&0&0&0&0\\ 1&0&1&0&1 \end{bmatrix} = \begin{bmatrix} 4&4\\ 4&4 \end{bmatrix}$$

R

$$\begin{bmatrix} 0&0&0&0&0&0\\ 0&2&2&2&2&0\\ 0&2&2&2&2&0\\ 0&2&2&2&2&0\\ 0&2&2&2&2&0\\ 0&0&0&0&0&0 \end{bmatrix} * \begin{bmatrix} 1&0&1&0&1\\ 0&0&0&0&0\\ 1&0&1&0&1\\ 0&0&0&0&0\\ 1&0&1&0&1 \end{bmatrix} = \begin{bmatrix} 8&8\\ 8&8 \end{bmatrix}$$

G

$$\begin{bmatrix} 0&0&0&0&0&0\\ 0&1&1&1&1&0\\ 0&2&2&2&2&0\\ 0&3&3&3&3&0\\ 0&4&4&4&4&0\\ 0&0&0&0&0&0 \end{bmatrix} * \begin{bmatrix} 1&0&1&0&1\\ 0&0&0&0&0\\ 1&0&1&0&1\\ 0&0&0&0&0\\ 1&0&1&0&1 \end{bmatrix} = \begin{bmatrix} 12&12\\ 8&8 \end{bmatrix}$$

B

Final image (add R, G, B columns).

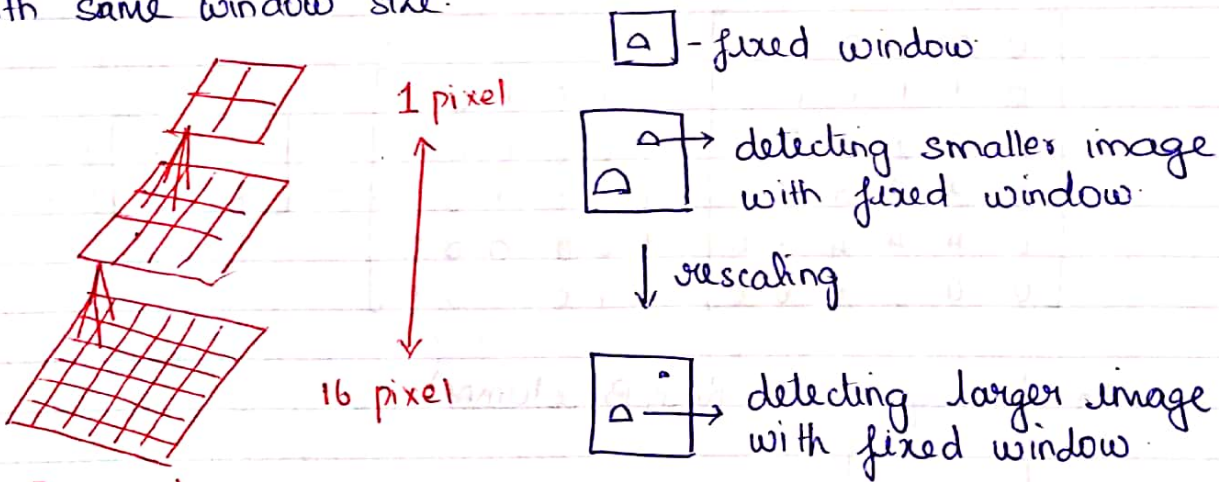
$$\begin{bmatrix} 24&24\\ 20&20 \end{bmatrix}$$

4. Template matching interpretation of convolution:

In the process of template matching, a template is given and it is found in a larger image. Similarly, a filter in a 2D convolution process can be imagined as a template that is swept across the image to find similar patterns. This is the template matching interpretation of convolution.

5. Multiple scale analysis with a fixed window size:

Multiple scale analysis can be achieved with pyramids. A filter with a fixed window size is selected and passed across the image. The filter initially determines objects of its own window size. To determine objects that are larger than the filter in the image, the original image is convoluted to a smaller scale. Thus now larger objects of the image are also determined by pyramids with same window size.



Pyramid

6. Compensation for spatial resolution loss using depth:

~~Flattening the image causes loss of spatial resolution.~~

Spatial dimensions decrease when the convolution iteration proceed ahead. Thus to compensate this loss, the depths are increased to retain as much as image information as possible. Thus this results in multiple layers. The number of co-efficients is maintained same to prevent loss of information when the height and width is reduced by increasing the depth.

7. Given:

$w = 128$

$k = 3$

16 conv. filters

To find:

Size of resulting tensor after conv. without zero padding.

Sol:

$$\text{Size} = \frac{w-k+2p}{s} + 1$$

$$= \frac{128-3}{1} + 1 \quad \text{(assuming stride = 1)}$$

$$= 125 + 1 = 126$$

Resulting tensor dimensions $\Rightarrow 126 \times 126 \times 16$

↳ filters

8. Prev. question with stride 2:

$w = 128 \quad k = 3 \quad S = 2 \quad p = 0.$

Resulting tensor size $= \frac{w - k + 2p}{S} + 1$

$= \frac{128 - 3 + 0}{2} + 1$

$= 125/2 + 1$

$= 63 + 1 = 64.$

Resulting tensor dimensions $= 64 \times 64 \times 16.$

9. Reducing no. of channels by 1x1 conv:

The dimension of a resulting tensor in a convolution depends on the no. of filters chose.

In a 1x1 convolution, the dimensions such as height and width are maintained ~~wheer~~ whereas the depth depends on no. of filters.

Thus choosing no. of filters lesser than the depth of image in a 1x1 convolution reduces the image channels.

10. **Interpretation of conv. layers.**

A convolution block consists of multiple convolution layers. Each layer has its own feature extraction tools and properties thereby to process the image.

Difference between deeper and early layers:

In a convolution base with multiple convolution layers, the early layers are more generalized. It performs low-level feature extraction. Whereas the deeper layers are more specific and perform high level feature extraction.

11. **Given:**

Image $\Rightarrow$ 4x4x3 image.

$$\underset{R}{\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}} \quad \underset{G}{\begin{bmatrix} 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}} \quad \underset{B.}{\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix}}$$

Max pooling with stride 2:

Sol.

$$\underset{R}{\begin{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} & \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \\ \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} & \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \end{bmatrix}} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} & \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} \\ \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} & \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$$

G

$$\begin{bmatrix} \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix} & \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix} \\ \begin{bmatrix} 3 & 3 \\ 4 & 4 \end{bmatrix} & \begin{pmatrix} 3 & 3 \\ 4 & 4 \end{pmatrix} \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 4 & 4 \end{bmatrix}$$

B

Final image (adding R, G, B channels)

$$\begin{bmatrix} 5 & 5 \\ 7 & 7 \end{bmatrix}.$$

12 **Purpose of pooling:**

Purpose of pooling is to down sample the input image representation. It helps in reducing the dimensions of the image, making it easy for processing and at the same time maintains the features of the image. It also serves as an advantage since there is no learnable parameters (weights). Further, it supports multiple scale analysis.

13. Data Augmentation & its use

Data augmentation is useful when the training datasets are less. In this process, the original dataset is increased by making distortions to the image and adding them to the dataset. These distortions made to the original images includes image rotation, dimension shift etc. to increase the no. of training images.

Data augmentation is applied only to the training set and not to validation (or) testing set.

14. Transfer learning:

Transfer learning is the process of using a pre-trained convolution network. In this process, the model is initially trained on a model, and is used in another problem of same type.

Thus the weights gained by the model on the priorly trained network can be used in the next model.

15. Freezing the co-eff of pre-trained n/w:

In the process of transfer learning, the pre-trained model is initially frozen. and added to the network.

By this way, the weights (parameters) of the original pre-trained model are preserved from any upgrades during back propagation.

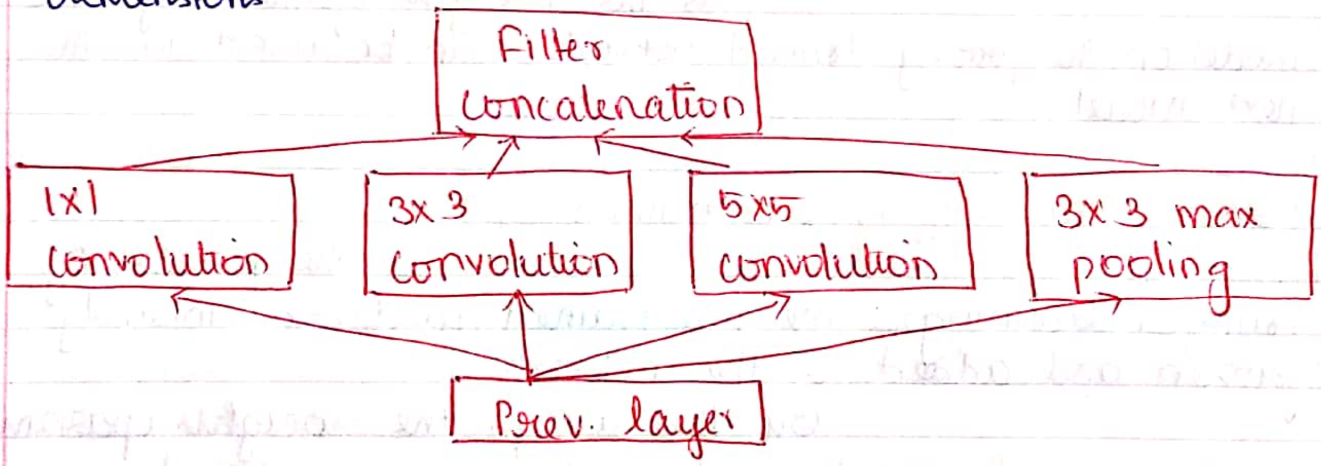
16. Fine tuning co-eff of pre-trained n/w:

In transfer learning, initially the pre-trained model is frozen. Finally, it is unfrozen for the purpose of fine tuning if needed.
It includes
1) Adding custom n/w on top of trained layers
2) Freezing trained layers
3) Training custom layers
4) Unfreezing layers in base n/w.
5) Jointly training the custom n/w.

17. Inception block:

~~Inceptive block~~ Inception block is used for multiple receptive field. These multiple receptive fields are then concatenated.

The purpose of this is to increase dimensions

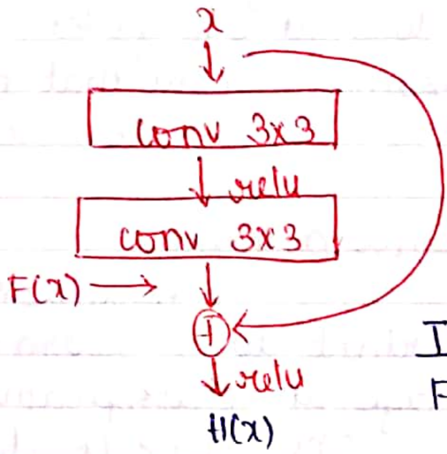


The output from prev. layer is passed to multiple receptive fields and then concatenated.

18 Advantages of residual field

Residual blocks have skip connections that by-pass the input to the last layer as well as give them to the first layer.

Due to this process, both the initial and final layers get trained easily and helps in creating deeper networks.



$H(x) = F(x) + x$.

It is easy to learn H(x) than F(x).

19. Intermediate activations of conv. layers.

It is possible to visualize what convolution networks do on images. Visualizing intermediate activations helps to check what filters do. Visualization of intermediate activations can be done by creating a new model from the existing model. This is done by using Model class instead of Sequential class. The Model class allows multiple outputs.

20. Visualizing filter weight of trained conv. layer:

The filter weights of trained convolution layers can also be vizualised by the process of ~~viz~~ visualization.

Filters are interpreted as templates that are being matched.

Using gradient ascent find input that maximize the response of the filter. (or) using gradient descent find input that will minimize loss at the filter.

Thus it vizualises input that minimizes loss (or) maximizes response.

21. Visualizing heatmap of class activation:

Visualizing heatmap of class activation is important to ~~f~~ determine which object to focus in an image in classification.

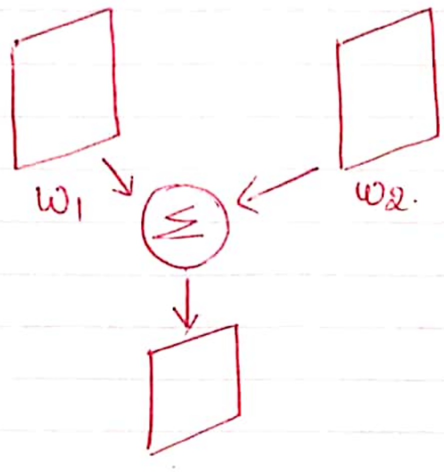
The classification depends on activation at the convolution layers (conv5).

High activations at the convolution layers is mapped to a specific image location.

'Conv 5' has multiple channels which should be combined to yield combined importance at each location.

Instead of averaging the channels, use a weighted sum that gives more importance to channels with higher gradients.

For the weight of each channel, the avg. gradient magnitude (pooled gradient) is used.



Applications:

Visualizing heatmap activations is used highly in

i) Document classification
ii) Sentiment analysis
iii) Author identification
iv) Question answering
v) Language translation.

Scanned with CamScanner