Vinta Reethu
Deep Learning for Vision
Indian Institute of Technology Hyderabad
Assignment-1

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Assignment-1

Vinta Reethu - ES18BTECH11028

February 4, 2021

1 Linear Filters

(a)

Given that

$$I = \begin{bmatrix} 2 & 0 & 1 \\ 1 & -1 & 2 \end{bmatrix} F = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}$$

Here we use zero padding to get the same dimensions for the output. Here I have padded the **I** matrix by adding zeros in the top row and left column.

$$I^{'} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 1 \\ 0 & 1 & -1 & 2 \end{bmatrix}$$

To perform the convolution operation we rotate the filter matrix **F** by 180°. Therefore, the rotated filter matrix is

$$F^{'} = \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}$$

$F * I^{'} =$

$$\begin{bmatrix} 0*-1+0*-1+0*1+2*1 & 0*-1+2*-1+0*1+0*1 & 0*-1+0*-1+0*1+1*1 \\ 0*-1+0*-1+2*1+1*1 & 2*-1+1*-1+0*1+-1*1 & 0*-1+-1*-1+1*1+2*1 \end{bmatrix}$$

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

Therefore, Convolution of I and F is

$$\mathbf{F}*I^{'} = \begin{bmatrix} 2 & -2 & 1 \\ 3 & -4 & 4 \end{bmatrix}$$

Vinta Reethu
Deep Learning for Vision
Indian Institute of Technology Hyderabad
Assignment-1

(b)

It can be seen that filter F is separable as

$$\begin{aligned} \mathbb{F} &= F_1 \cdot F_2 \\ &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & -1 \end{bmatrix} \end{aligned}$$

Rotating the matrices by 180° we get,

$$\begin{aligned} F_1' &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ F_2' &= \begin{bmatrix} -1 & 1 \end{bmatrix} \end{aligned}$$

First we pad the image **I** with one row on the top.

$$I' = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 0 & 1 \\ 1 & -1 & 2 \end{bmatrix}$$

$$\begin{aligned} F_1 * I' &= \begin{bmatrix} 0+2 & 0+0 & 0+1 \\ 2+1 & 0+-1 & 1+2 \end{bmatrix} \\ &= \begin{bmatrix} 2 & 0 & 1 \\ 3 & -1 & 3 \end{bmatrix} \end{aligned}$$

Let's pad this $F_1 * I'$ matrix by adding one zeros column to left.

$$(F_1 * I')' = \begin{bmatrix} 0 & 2 & 0 & 1 \\ 0 & 3 & -1 & 3 \end{bmatrix}$$

$$\begin{aligned} F_2 * (F_1 * I')' &= \begin{bmatrix} 0+2 & -2+0 & 0+1 \\ 0+3 & -3-1 & 1+3 \end{bmatrix} \\ &= \begin{bmatrix} 2 & -2 & 1 \\ 3 & -4 & 4 \end{bmatrix} \end{aligned}$$

(c)

Using the expansion of I*F to solve this,

$$
\begin{aligned}
(I * F)[m,n] &= \sum_k \sum_l I[k,l] F[m-k, n-l] \\
&= \sum_k \sum_l I[k,l] \left(F_1[m-k] F_2[n-l]\right) \\
&= \sum_l F_2[n-l] \sum_k I[k,l] F_1[m-k] \\
&= \sum_l F_2[n-l] \left(I * F_1\right) \\
&= \sum_l F_2[n-l] \left(F_1 * I\right) \ \{\text{Associativity of Convolution}\} \\
&= (F_1 * I) * F_2 \ \{\text{Associativity of Convolution }\} \\
&= F_2 * (F_1 * I)
\end{aligned}
$$

Hence proved.

(d)

Multiplications in part (a) is **24**.
Multiplications in part (b) is **24**.
Both of them take equal number of multiplications.

(e)

i. $M_1 \times N_1 \times M_2 \times N_2$ multiplications are needed to do direct 2D convolution.
ii. $(M_1 \times N_1 \times M_2)+(M_1 \times N_1 \times N_2) = (M_1 \times N_1 \times (M_2+N_2))$ multiplications are needed to do direct 1D convolution.
iii. The only term that is different in both of them is $M_2 \times N_2$ **,** (M_2+N_2). For very large values of M_2, N_2**;** $(M_2+N_2) < M_2 \times N_2$. Hence it is more efficient to perform two successive 1D convolutions than one 2D convolution.

2 Canny Edge Detector

(a)

Initially the magnitudes of derivatives will be

$$D = \sqrt{D_x^2 + D_y^2}$$

As the point is on X-axis, $D_y = 0$

$$D = D_x$$

After rotating the edge by θ the coordinates become

$$x' = x\cos\theta$$
$$y' = x\sin\theta$$

The new magnitude is given by

$$\begin{aligned} &= \sqrt{D_{x'}^2 + D_{y'}^2} \\ &= \sqrt{\cos^2\theta D_x^2 + \sin^2\theta D_x^2} \\ &= D_x \end{aligned}$$

Hence the rotated edge will be detected by canny edge detector. It is rotation invariant.

(b)

If long edges are broken into short segments separated by gaps, this means that the low threshold is high as a result we are loosing some weak edges. To avoid this we should decrease low threshold. Some spurious edges appear when high threshold is low. To avoid this we can simply increase high threshold.