

Lab-10-1 Convolution

- Convolution: 이미지 위에서 stride 값 만큼 filter(kernel)를 이동시키면서 겹쳐지는 부분의 각 원소의 곱을 곱해서 모든 더한 값을 출력으로 하는 연산

- stride: filter를 한 번에 얼마나 이동할 것인가

- padding: zero-padding

- `Pytorch.nn.Conv2d(in_channels, out_channels, kernel_size, stride=1, padding=0, dilation=1, groups=1, bias=True)`

$$\text{out}(N_i, \text{Out}_j) = \text{bias}(\text{Out}_j) + \sum_{k=0}^{C_{in}-1} \text{weight}(\text{Out}_j, k) * \text{input}(N_i, k)$$

Cross-correlation!

input shape = (N x C x H x W)

(batch_size, channel, height, width)

$$\text{Output size} = \frac{\text{input size} - \text{filter size} + (2 * \text{padding})}{\text{stride}} + 1$$

ex) `Conv = nn.Conv2d(1, 1, 1, stride=4, padding=0)`

`inputs = torch.Tensor(1, 1, 22, 22)`

`out = Conv(inputs)`

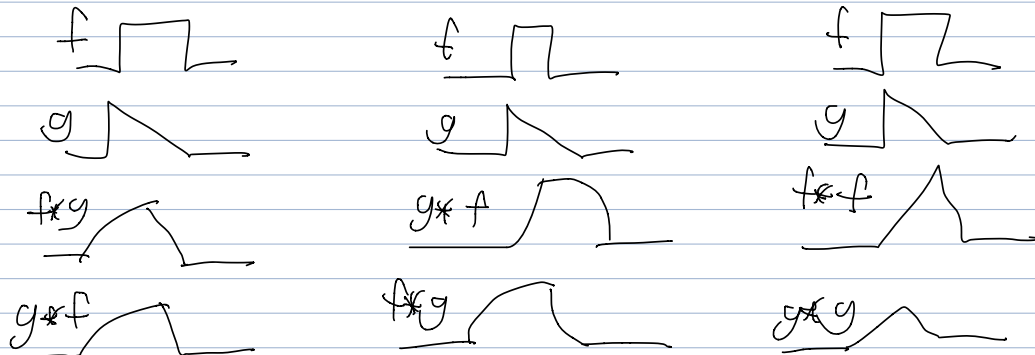
`out.shape`

`>> torch.Size(1, 1, 5, 5)`

- max pooling: 범위 안에 큰 값
- average pooling: 범위 안의 평균

torch.nn.MaxPool2d(kernel_size, stride=None,
padding=0, dilation=1, return_indices=False,
ceil_mode=False)

• Convolution Cross-correlation Autocorrelation



$$(f \otimes g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau) g(t-\tau) d\tau = \int_{-\infty}^{\infty} f(t-\tau) g(\tau) d\tau$$

첫 번째 계산을 Convolution, 두 번째 계산을 Cross-correlation