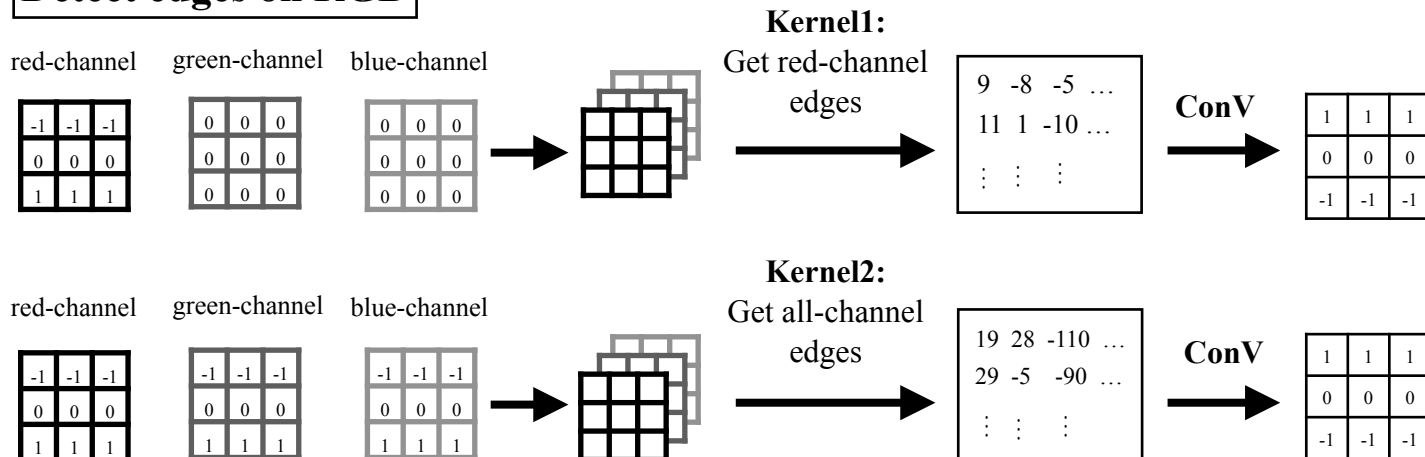


SC201 Lecture 14

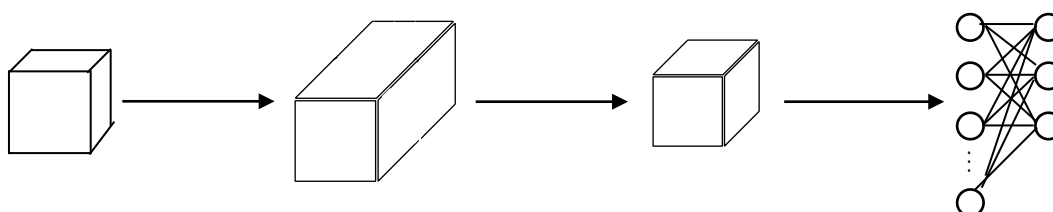
Detect edges on RGB



Input	Filter	Padding	Stride	Output

若使用8個3x3x3的filters, padding=1, stride=1 ,

對3x32x32的RGB影像convolve , output = _____



SC201 Assignment5.ipynb

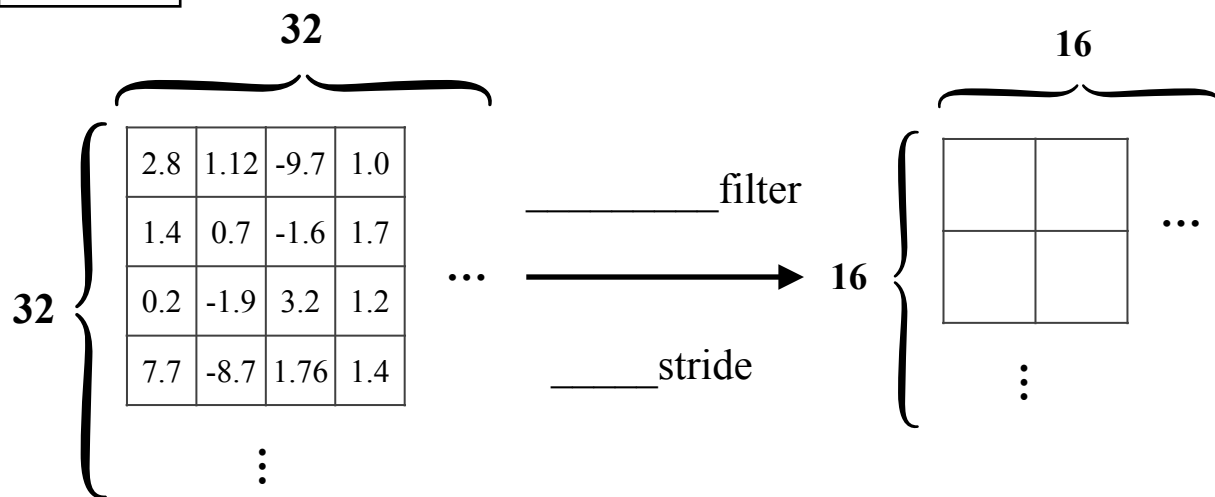
```

model = _____ (
    # N x 3 x 32 x 32
    nn.Conv2d(in_channels = _____, out_channels = _____, kernel_size = _____, padding = _____),
    _____,
    # N x 32 x 32 x 32
    nn.Conv2d(in_channels = _____, out_channels = _____, kernel_size = _____, padding = _____),
    _____,
    # N x 16 x 32 x 32
    _____,
    _____,
    _____
)

```

optimizer = optim.SGD(model.parameters(), lr=learning_rate, momentum=0.9, nesterov=True)

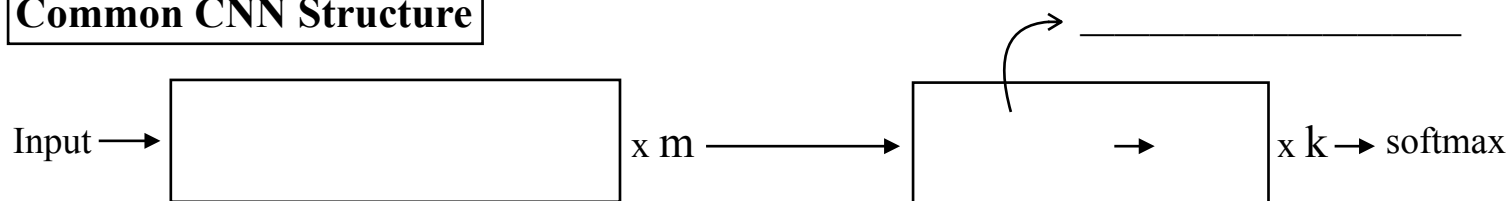
MaxPool



讓電腦在每一個 channel 找出最重要的 _____ (去蕪存青)

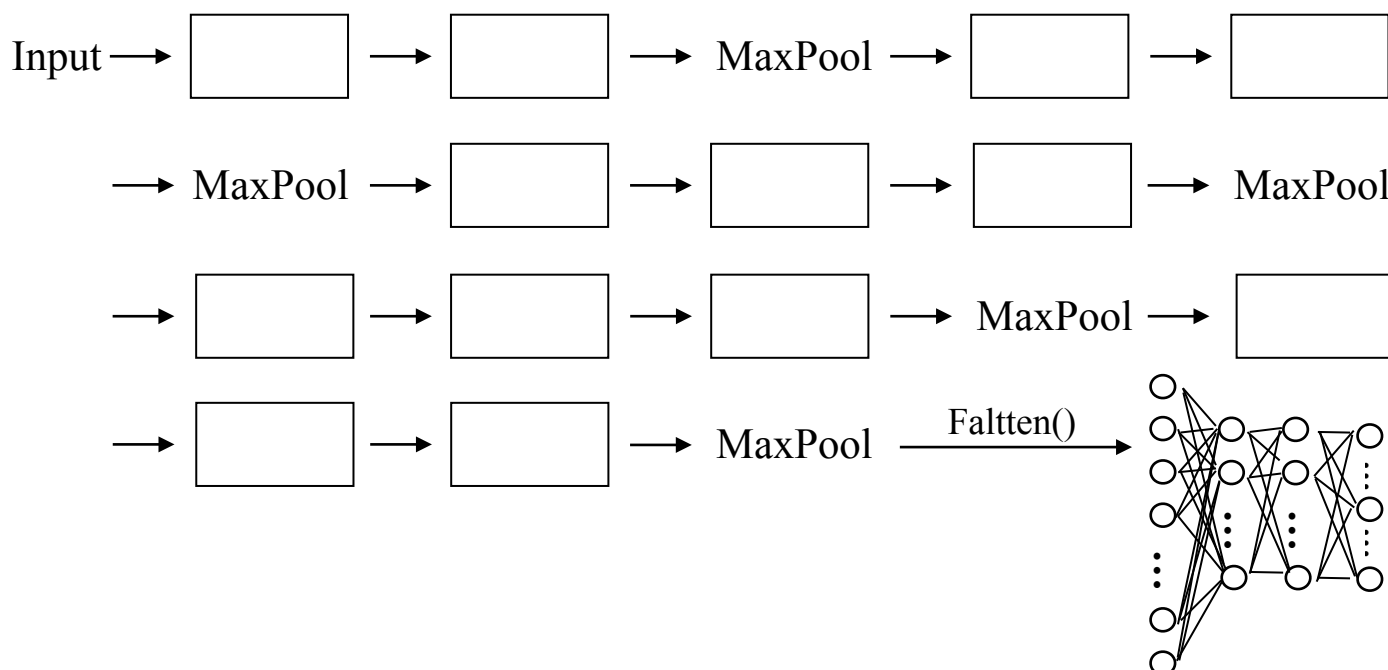
_____ (kernel_size=2, strider=2)

Common CNN Structure

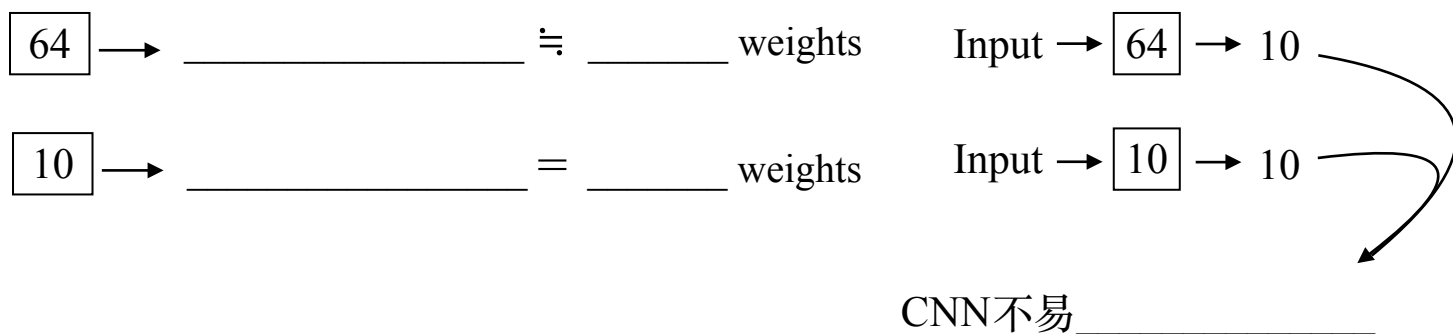
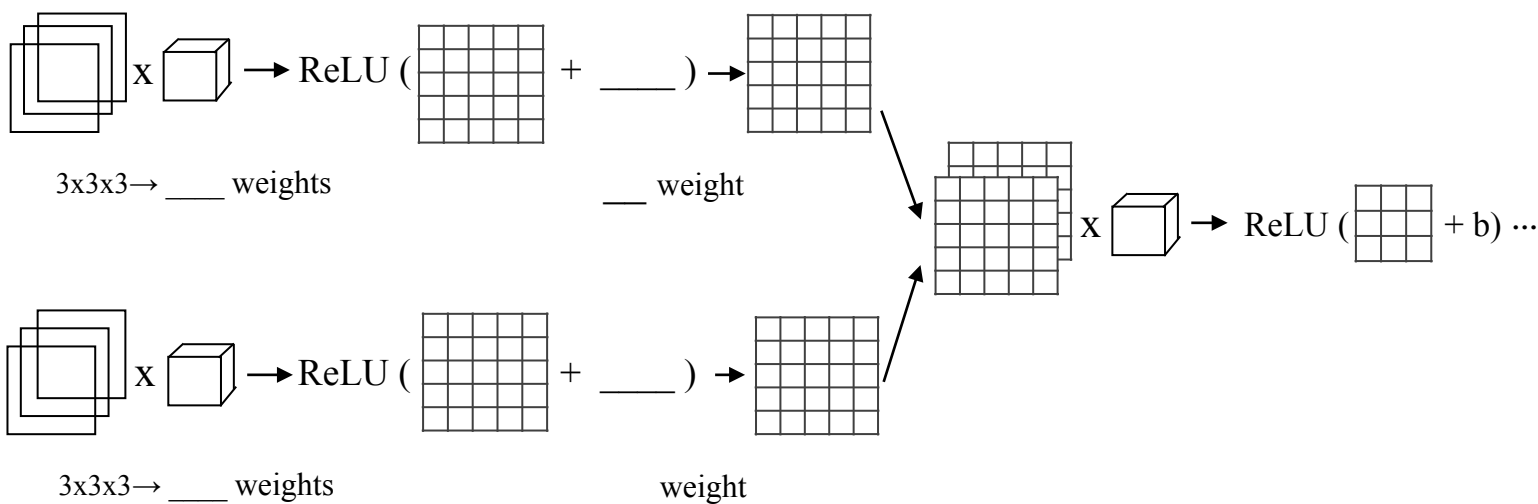


VGGNet

- 2014
- _____ Accuracy on ImageNets (14,000,000 with 1,000 classes)
 - Human

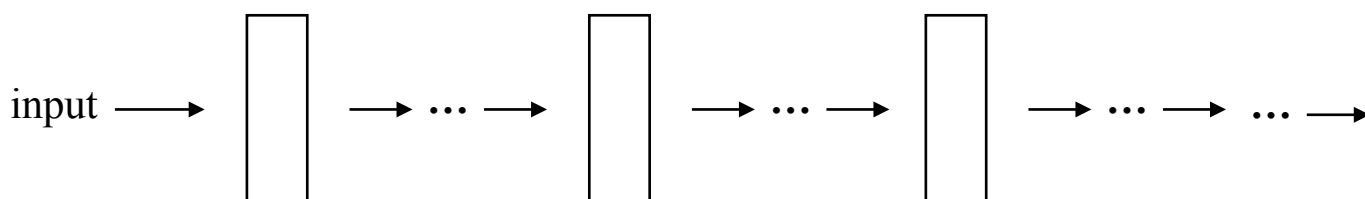


Number of parameters on CNN

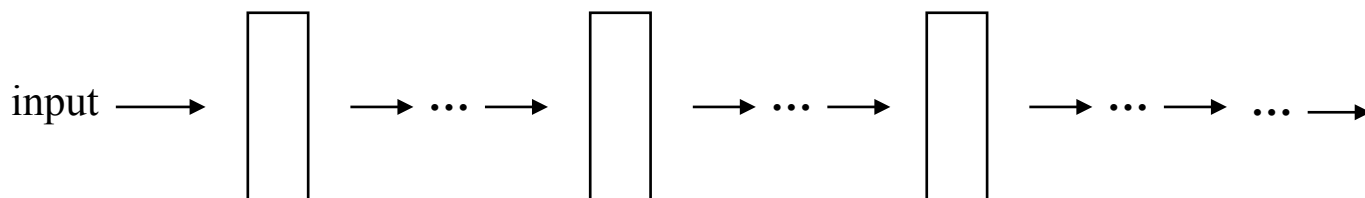


BatchNorm 2015

< Internal Covariate Shift > # 把每一層 output 當成 input \rightarrow _____



< Batch Normalize > $\frac{(x - \mu)}{\sigma}$ # Before _____



PyTorch

< Data Pre-processing >

```
import torchvision._____ as T
```

```
transform = T.Compose([ _____, T.ToTensor(), T.normalize(mean=mean, std=std)])
```

< Load Data >

```
import torchvision._____ as dset
```

```
train = dset.ImageFolder( _____ , transform=transform)
```



```
List[Tuple(Tensor, int)]
```



< Create Mini-batches >

```
from torch.utils.data import _____
```

```
mini_trains = DataLoader(train, batch_size=BATCH_SIZE, shuffle=True)
```