# 1 Structure of CNN

In this tutorial, we will employ CNN to solve the classification problem. First, we will give a brief introduction about MNIST dataset. Next, the convolutional layer, pooling layer, flatten layer, and fully connected layer will be introduced.

## 1.1 MNIST Dataset

MNIST Dataset is a database of handwritten digits, which has a training set of 60,000 examples, and a test set of 10,000 examples. Each example in the MNIST is a 28 × 28 image.

In this section, we will choose 2,000 images as our training set, and 500 images as our test set. The shape of the training and test set will be:

- X_train.shape = (2000, 784) → (2000, 1, 28, 28)
- X_test.shape = (500, 784) → (500, 1, 28, 28)
- y_train.shape = (2000,1) y_train_ohe.shape = (2000,10)
- y_test.shape = (500, 1) y_test_ohe.shape = (500,10)

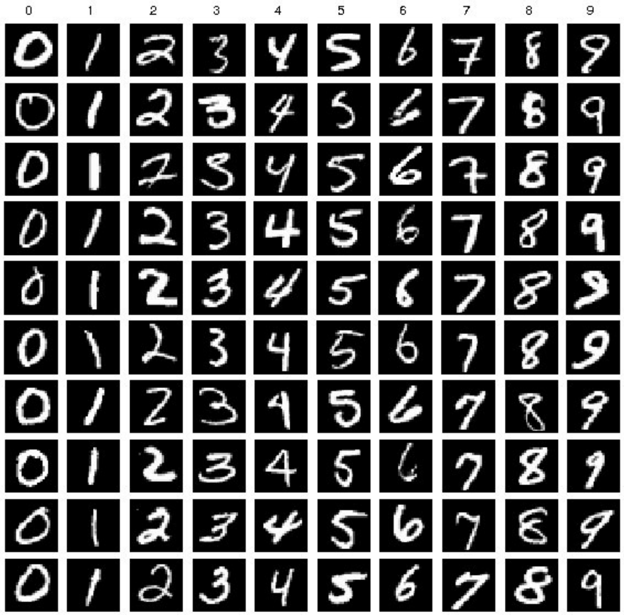

Figure 1: MNIST Dataset.

## 1.2 Convolutional Layer

Assume the size of the input image is $H \times W$, then the size of the output image after passing the convolutional layer will be

$$\begin{aligned} H' &= \frac{H - F_H + 2P_H}{S_{\text{conv}}} + 1 \\ W' &= \frac{W - F_W + 2P_W}{S_{\text{conv}}} + 1 \end{aligned}, \tag{1}$$

where $F_H$ is the height for filter, $F_W$ is the width for filter, $S_{\text{conv}}$ is the stride size, and $P$ is the padding size.

- Filter: Weights. In the training process, filters need to be updated.

- Stride is the number of pixels shifts over the input matrix. When the stride is 1 then we move the filters to 1 pixel at a time. When the stride is 2 then we move the filters to 2 pixels at a time and so on
- Padding is simply a process of adding layers of zeros to our input images

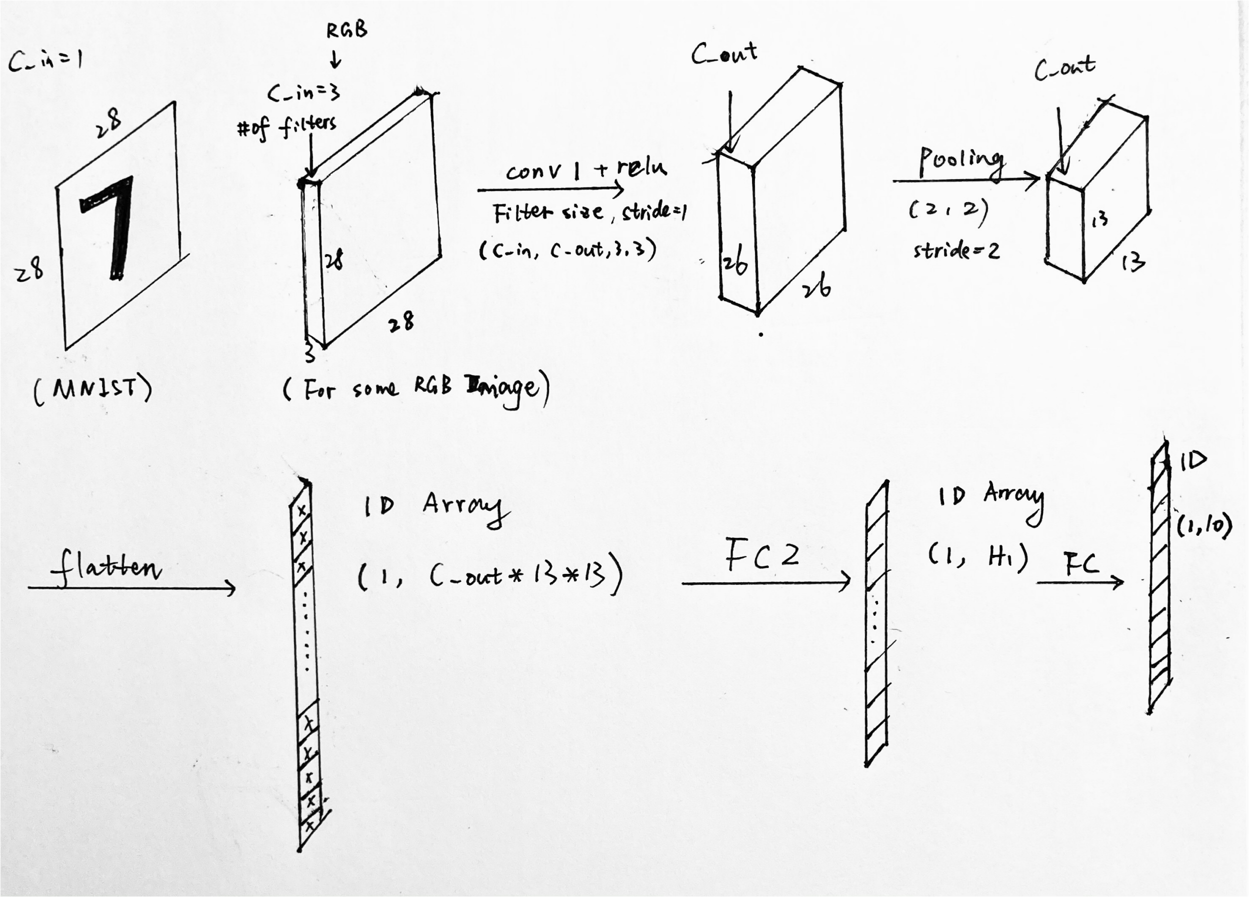

Figure 2: Illustration of 2D CNN. Assume the number of samples in the training set is $N$, then for MNIST dataset, the input will be a 4D array ($N$, $\mathrm{C_{in}}$, 28, 28). The training process will be: ($N$, $\mathrm{C_{in}}$, 28, 28) $\xrightarrow{\text{conv1}}$ ($N$, $\mathrm{C_{out}}$, 26, 26) $\xrightarrow{\text{pool1}}$ ($N$, $\mathrm{C_{out}}$, 13, 13) $\xrightarrow{\text{flatten}}$ ($N$, $\mathrm{C_{out}}$ * 13 * 13) $\xrightarrow{\text{FC2}}$ ($N$, $H_1$) $\xrightarrow{\text{FC}}$ ($N$, 10)

## 1.3 Pooling Layer

Assume the size of the input image is $H \times W$, then the size of the output image after passing the pooling layer will be

$$\begin{aligned} H' &= \frac{H - \text{Poolsize}_H}{S_{\text{pool}}} + 1 \\ W' &= \frac{W - \text{Poolsize}_W}{S_{\text{pool}}} + 1 \end{aligned}, \tag{2}$$

## 1.4 Flatten Layer

Flatten 3D array to 1D array

## 1.5 Fully connected Layer

Fully connected layer is actually a hidden layer in the ANN.