**Project #2: CNN Architecture Implementation**

2019170540 Wonjun, Oh (오원준)

**1. Description of your code**

**Residual Block** The key to this section is to adjust the output size of the model, depending on whether it is Downsample or not. If Downsample is True, halve the image size by setting it to Stride 2, Padding 1 on the Conv 3x3 layer. If the Downsample is False, the output image size must be the same as the input, so Conv 3x3 is set to Stride 1, Padding 1.

**Layer1** Take 3 channels of input and give 64 channels of output. At this time, the image size is halved in Conv of kernel\_size 7, stride 2, and padding 1, changing from 32X32 to 16X16. Next, 2x2 pooling changes output image size from 16x16 to 8x8.

**Layer2** It consists of a total of three Residual Blocks. The input of each block is the same as the previous output, and the last layer sets Stride to 2 to halve the image size. Image size changes from 8x8 to 4x4.

**Layer3** It consists of a total of four Residual Blocks. The input of each block is the same as the previous output, and the last layer sets Stride to 2 to halve the image size. Image size changes from 4x4 to 2x2.

**Layer4** It consists of a total of six Residual Blocks. The input of each block is the same as the previous output, which keeps the image size from 2x2 to 2x2, as described in the problem.

**Average Pooling** Halve the image size with 2X2 Pooling. Image size changes from 2x2 to 1x1.

**Fully Connected Layer** Flatten the result of passing through the pooling layer, then pass it through the fully connected layer. In this case, the output is 10 equal to the number of classes in CIFAR-10.

**2. Results**

Only 1 Epoch was trained for the completed code, and 81.05% Accuracy was finally obtained.

텍스트, 폰트, 스크린샷, 타이포그래피이(가) 표시된 사진

자동 생성된 설명

(Figure 1) Results

**3. Discussions**

**Differences from Original ResNet-50** The original structure consists of a total of five layers (in variable names), one more for each of our models. At this time, the last layer of the original model is to reduce the image size once more, so pooling consists of 1x1. However, since our model has one fewer layers, the last 2x2 pooling layer performs the downsizing that was previously performed in Layer 5.

**Very small input size** The total amount of information that the model can analyze is the total amount of information that the maximum image has. That is, the larger the image size, the more information the model can analyze. Our model performs classification for very small sized Inputs, so we can see that we classify them well even with quite little information.

**End.**