Here are what we have done so far. The first and the most essential thing that we have done is construct the framework of SRCNN which includes three parts: prepare data, SRCNN algorithm and error analyze. Then we continued to work on those three parts. We formalized our training dataset to fit the model. Also, we create a function to implement SRCNN. Finally, we found a better loss function which is used to analyze the error and improve our algorithm.

The central idea of SRCNN algorithm is the three-layer convolution. The first layer is patch extraction and representation. This layer extracts patches from the low-resolution image. It can be done by convolving the image by n1 filters. After this step, we can have n1 feature maps of low-resolution image. The second layer transforms the n1-dimensional vectors into n2-dimensional vectors through a nonlinear operation. Finally, the last step convolves the n2 feature maps of high-resolution image into the final full image.

Compared to the other interpolations, SRCNN uses deep learning to fill in the missing details in the upscaled image, which is more reasonable and also can receive a better result.

Until so far, we have simply implemented those three convolutions in the SRCNN by using tensorflow. In the future, we need to find better parameters such as filter numbers in the first two layers. The more filters we use in the convolution, the better results will be. However, too many filters will dramatically influence the speed of the algorithm. Thus, we will try different n1 and n2 to find better filter numbers. Moreover, different layers have different weights and biases for convolution, we also need to do experiments on finding the best ones.

After completing the SRCNN algorithm, we will use the pre-processed data to train the model and use loss function to evaluate our model. Next, welcome Shuo to introduce the loss function.