Super-resolution for Remote Sensing Images

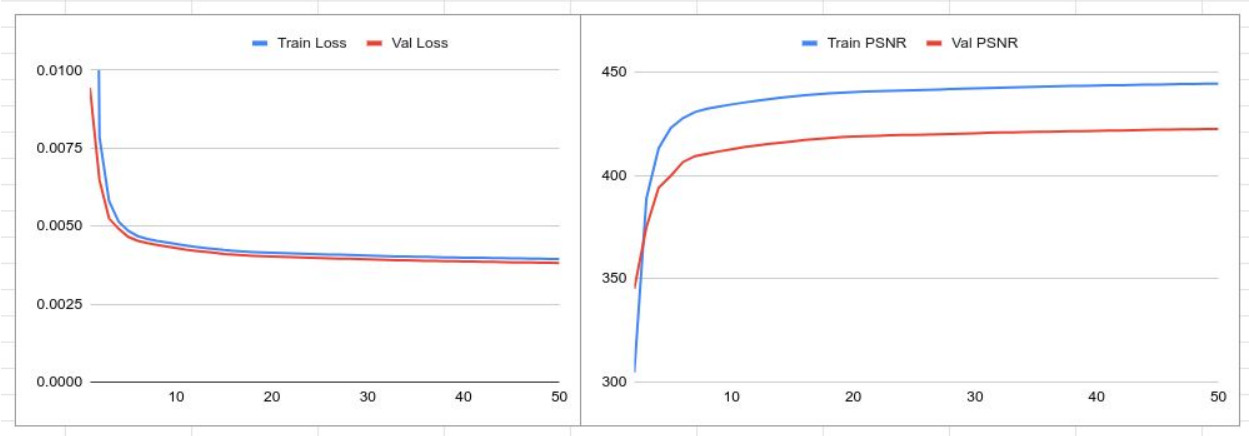
Code location : https://github.com/anandcu3/sr-cnn-remote-sensing

Methods & Experiments

The UC Merced Land Use Dataset was downloaded from their website and the data is split into a 70:10:20 split for train, validation and testing purposes. The images were 256*256 images usually and were resized to this size in the cases when the sizes were off by a few pixels. These images were then reduced by a scaling factor of 4 to a 64*64 image. These lower images were then resized back to the original (256*256) size using bipolar interpolation. These are used as the input to the SR-CNN. The SR-CNN architecture was built using Pytorch. Here is the architecture used.

```
SRCNN(
  (conv1): Conv2d(3, 64, kernel_size=(9, 9), stride=(1, 1), padding=(2, 2), padding_mode=replicate)
  (conv2): Conv2d(64, 32, kernel_size=(1, 1), stride=(1, 1), padding=(2, 2), padding_mode=replicate)
  (conv3): Conv2d(32, 3, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2), padding_mode=replicate)
)
```

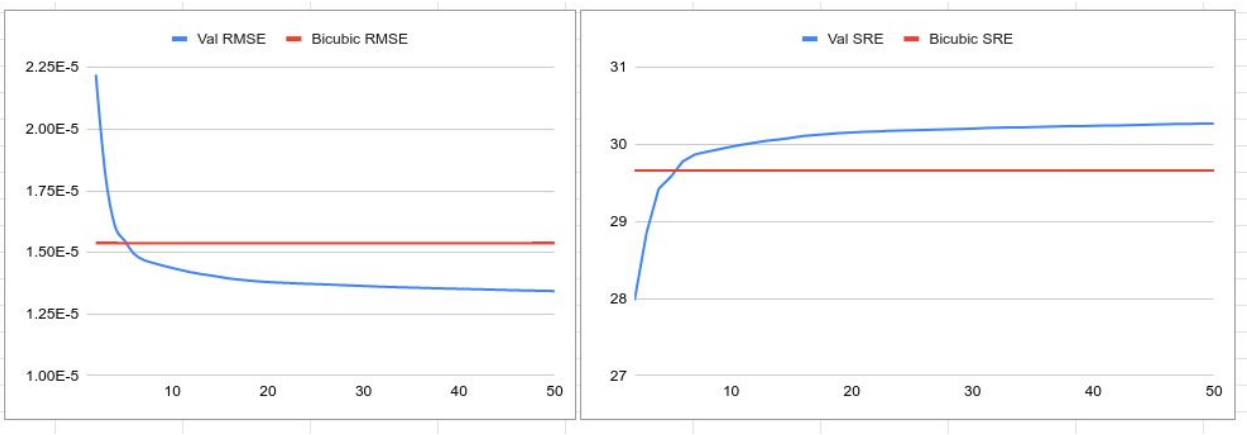
During training, the MSE Loss was monitored over the 50 epochs used to train the CNN. Also PSNR (Peak Signal to Noise Ratio) was monitored. Validation and training values are in agreement through the training procedure which is a good sign. Also the network trains very quickly and it would be possible to increase the complexity of the network in the future because sufficient data is available for a bigger network.



Results and Evaluation

RMSE (Root Mean Squared Error) and SRE (Signal to Reconstruction Error ratio) were monitored across 50 epochs for the validation set. Here is a graph with the RMSE and SRE values across various epochs. As we can see from the graph the SRCNN beats the Bicubic

interpolation method at around 6th epoch.



The same metrics were calculated for the test set as well. Here are the test Set Results:
Bicubic RMSE : 1.572e-05
Bicubic SRE : 29.843
SRCNN RMSE : 1.382e-05
SRCNN SRE : 30.442

Here are the sample results from the test set. For each image the left image is the original image, the center is the bicubic interpolation generated image, right image is the output from the network.

Next Steps

To improve accuracy, steps like hyperparameter optimization can be done. Modifying the network architecture slightly by modifying the kernel sizes of the convolution can also help. Furthermore a simple CNN might not be the best network for this purpose. A network like GAN (Generative Adversarial Network) or a VAE (Variational Autoencoder) might be better suitable and experiments using these can be conducted.

Other relevant work