Upscale Resolution on Multi-Frame Remote Sensed Images

Ahmet Şahin Yener Deniz Zağlı

Abstract

The importance of satellite images is increasing day by day. The biggest reason for this increase in importance is that satellite images contain important geographical information about the world. Many scientific researches about the world have been done with these geographical information and it is still being carried out. It not only gives information about the surface, but also allows us to investigate and predict weather events. As in the investigation of the structure of the transfer events, easy observation in places where it is difficult to observe with other possibilities other than satellite, in examining the possible effects of natural disasters and in estimating the possible effects, in examining the dynamic structure of the atmosphere and in obtaining information about it, in the majority of military systems, in mapping the desired region. Satellite images are used in many areas. Due to the increasing importance of satellite images, some research centers, some states and some private companies have their own satellites to acquire satellite images. Since high resolution satellite images have high financial value, such institutions and organizations are working to improve the quality of the satellite images they obtain. Increasing the resolution of the satellite images obtained is an important research subject today. We aimed to increase the resolution of these satellite images obtained in our Upscale Resolution on Multi-Frame Remote Sensed Images project. Our goal is to create a new satellite image with super-resolution by integrating low resolution satellite images. As Dataset, we use satellite images taken by PROBA-V satellite. There is a dataset collected by the European Space Agency using PROBA-V satellite. The standard satellite image resolution within this dataset is 300m. The resolution of high-resolution satellite images is 100m. During the training phase of the model we created, we applied super-vised learning style. The model we created was trained by fusing the satellite images with a resolution of 300m and comparing the result with satellite images with a resolution of 100m. In this way, we realized the satellite image with a high cost of 100m resolution by fusing satellite images with a resolution of 300m. In the Upscaled Resolution on Multiframe Remote Sensed Images project, we used Conv3D and WDSR architectures to obtain satellite images with a resolution of 100m.

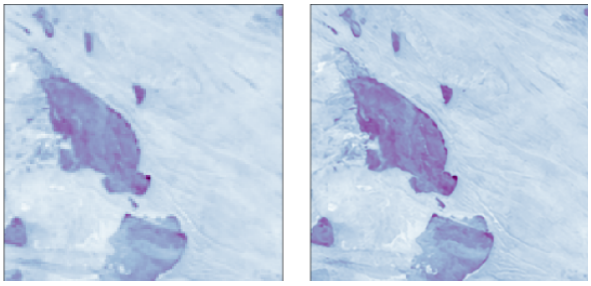

Figure 1. Low Resolution - Super Resolution

1. Introduction

Remote sensing at it's heart carries a trade-off. This trade off is that no single satellite is capable of achieving high temporal resolution, spatial resolution and spectral resolution at the same time. [1]. One way to provide high spatial resolution and temporal resolution at a single time is using [2] satellite constellations. Satellite constellations are satellite arrays on LEO (low earth orbit), which provides high spatial resolution and low amounts of atmospheric disturbance while the revisit times for a single ground scene for a single satellite times are much higher then MEO (medium earth orbit) and GEO (geostationary earth orbit), since there are multiple satellites in a constellation it can also provide high temporal resolution earth imagery [3]. However the cost of LEO missions, and the cost of operating and launching a satellite constellation bars most public research being done on high spatial and high temporal resolution satellite imagery. Landsat-8 is an open to public satellite imagery having medium spatial resolution (30mx30m=1 pixel) it suffers from low temporal resolution, providing an image of the same ground scene with 16 day intervals[4]. Terra and Aqua Satellites provide open to public imagery at high revisit times, together providing imagery of a single ground scene with 12 hours of interval, while it suffers from coarse spatial resolution. [5] Work on spatio-temporal fusion of low

spatial resolution and high temporal resolution with is an active research area [6]. The task we are trying to achieve is based on similar situation and has been worked on in Computer Vision before with deep neural networks and algorithm based approaches.[7][8][9] Our focus is mainly on deep learning approaches

2. Method

2.1. Model

Data fed into the machine is Low-Resolution frames of size (128 x 128). For each high resolution ground truth 7 low resolution images are selected of the scene. We rank the low-resolution images with the provided mask, according to the amount of clear-pixels observed in the low-resolution scene imagery. We set the lower boundary to be 0.15 of pixels to be dirty at most. We stack the images to the reference image. We also extract a mean image out of the image stack and use it as the residual path of the network which changes the bicubic upsampling operation in the figure 2.

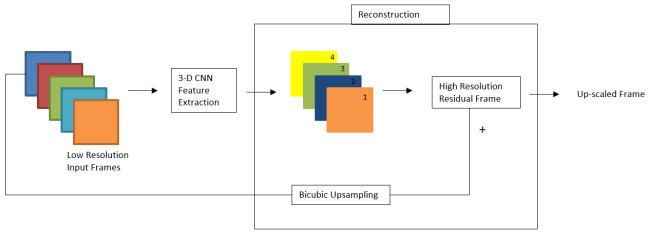

Figure 2. 3DSRnet

The architecture proposed in figure 2 is the general structure of the network for the super-resolution task. The changes applied are we change the feature extraction by 3D Convolution to wdsr-b (figure 3) blocks for feature extraction. Also the residual path is changed to a 2D Convolution applied to the mean images obtained from the image stack.

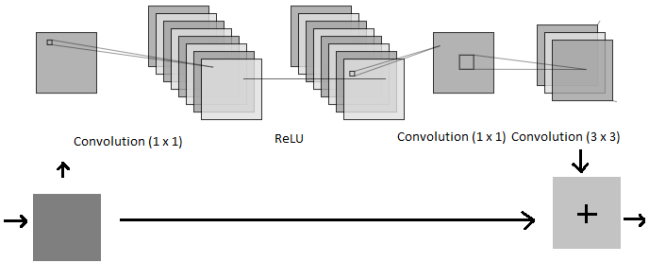

Figure 3. WDSR-B

A wdsr-b block is composed of 3 units. Here is the modified inner workings of a wdsr-b block explained.

- Padding is applied with zeros over the image to keep the original width and height of the incoming image.

- The expansion unit applies a linear low-rank convolution (1x1) over the image batch, increasing the channel size by 6.

- ReLU [10l activation function is applied over the output of the expansion unit

- The decay unit lowers the channel-size also with a linear low-rank convolution (1x1) over the activated output. The ratio of this was chosen to be 0.8 in this work.

- The normalizing unit takes the output of the decay unit and lowers it to be the channel size matching the input to the block with a convolution with (3x3) kernel size

- The final output to the wdsr b block is the input added with the convolution output.

We stack 8 wdsr-b blocks one following each other with 32 filters per block.

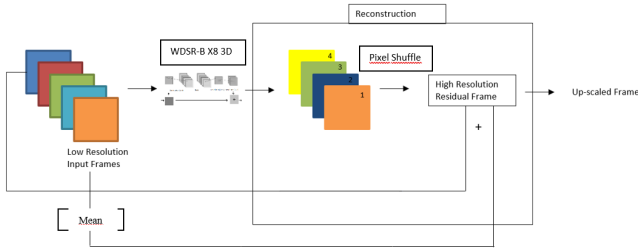

Figure 4. Our Architecture

The regularization is performed with weight normalization for each convolutional filter. The weight normalization is picked for batch normalization has been showed to be a too strong of a constraint for the task of super-resolution. Instance normalization however depends too strongly on the incoming patches which shows variance greater than what is expected in changing satellite imagery. We've found variances on low-resolution satellite imagery and the features extracted out of them to be higher than of those of natural images such as in ImageNet dataset [11]. We use a reshaping operation that is known as pixel shuffle. Pixel shuffle [12] 2 takes in inputs of form C*r2̂,H,W and then reshapes it into C,H*r,W*r for each element in the batch. This makes the output after the network flow which is of high resolution residuals to into its final shape.

2.2. Loss Function

The output of the network is took and masked with the valid pixel mask present for each high resolution image. Since

drifts of 50m is possible in the original satellite imagery, which constitutes a 6 pixel shift in either direction we compensate for this shift in this way. We calculate 49 different losses (7x7) moving in either way, and at the end we pick the lowest loss recorded. The loss function is a simple l1 loss subtracting each pixel of the valid pixels from the prediction with the ground truth.

While a patching operation was performed with 4 patches per image to 16 patcher per image the RAM capabilities of the google colab was over-run which is 25 GB . That means we did not perform any patching of the images. We extrapolate that the total capability required is 36 GB. While the dataset could be reduced we decided against that since the shifts in the image are mostly uniform. Losing %30 percent of the training dataset would perform worse than the amount of missed pixels that would be fixed by the patching operation and since convolution expensiveness was less of a bottle neck in the colab system then it is of ram we have used no patching.

3. Experimental Settings

3.1. Dataset

As Dataset, we determine the specific regions of the European Space Agency and use the dataset with satellite images of those regions. Satellite images of these regions determined by European Space Agnecy were taken by PROBA-V. There are only RED and NIR (Near Infared) bands in this dataset. Satellite images actually consist of more bands. For example, the satellite image of Landsat-8 consists of 11 bands. In our dataset, only RED and NIR (Near Infared) are available. We trained the model we created only with the data in the NIR (Near Infared) band. We made our predictions with the same band data. Satellite images in Dataset are completely gray-scale. The data in Dataset are also divided into two according to their resolution. The resolution of low resolution satellite images is 300m x 300m. The pixel count of the low resolution satellite image is 128x128. The resolution of high resolution satellite images is 100m x 100m. The pixel count of high resolution satellite images is 384 x 384. One of the biggest problems of satellite images is the weather effect. The most common problem among weather effects is cloud cover. To get maximum efficiency from a satellite image, at least 70% of the pixels must be visible. So for maximum efficiency, cloud cover should be maximum 30%. In order to solve this problem, more than one satellite images belonging to the same region were collected in the satellite images of PROBA-V satellite collected by the European Space Agency. These satellite images collected from the same region are recent. From the satellite images in Dataset, those starting with HR represent High-Resolution, and those starting with LR represent Low-Resolution data. NIR (Near Infared) data that we can use in Datasette are 1450 in total. Data that can be used for train and test are reserved in Dataset. As a result of this distinction, 1160 data are used for the train and 290 data are used for the test. For example, the satellite image with high resolution is shown in figure 5, and the satellite image with low resolution is shown in figure 6.

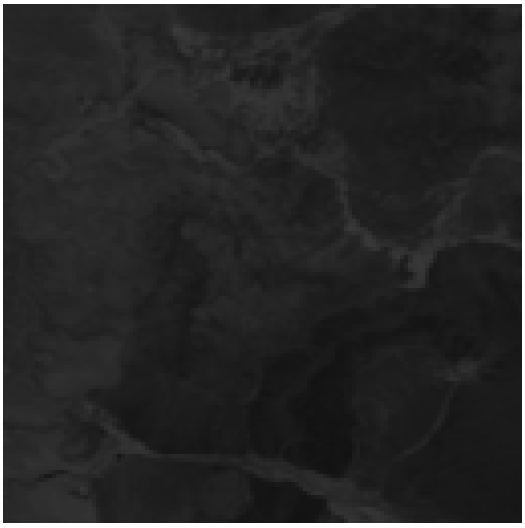

Figure 5. Low Resolution Satellite Image

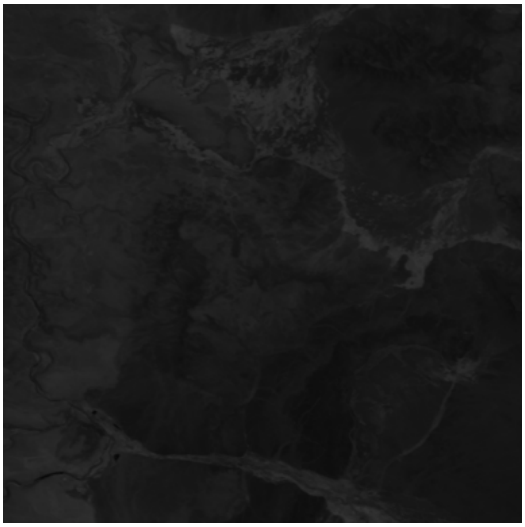

Figure 6. High Resolution Satellite Image

3.2. Pre-Processing

Satellite data are larger in size compared to other visual data. Especially if satellite data contains geotag, its size increases even more. Because of this situation, it takes a long time to convert satellite images to array format. For this reason, we converted all the satellite images that we will use for dataset to numpy array format and saved these numpy arrays

that we converted in npy format. This stage takes about 3 hours. As a solution to the cloud cover effect in the Dataset section, we mentioned that there are more than one satellite images belonging to the same region. First, we keep all these satellite images of the same region in a single array and create a patch. Then we calculate the dirty pixels by checking the images in the patch we create one by one. Dirty pixels are pixels that do not appear due to the cloud we call. If the number of dirty pixels is 15% or more than 15% in the entire satellite image, we delete this satellite image from the patch we created. Then we sort all the images according to the dirty pixel percentage. From low percentage of dirty pixels to high percentage of dirty pixels. Then we take the first k of the satellite data in the path according to the k value we give and delete the other satellite images. We have determined 7 as the k value. The value 7 is suitable for all values in the dataset. As at the beginning of the pre-process, we recorded the patches containing 7 satellite images, which we have chosen again, in a file called patch. After these pre-process steps, the new dataset model we have created from dataset is ready for the training.

3.3. Evaluation metrics

We used PSNR (Peak Signal to Noise Ratio) as the evaluation metric. The purpose of PSNR (Peak Signal to Noise) is a measure of how much the original image is measured from noise. When doing this calculation, we first calculate the peak value. This peak value is 255 for satellite images in our dataset. Then we square the difference between the pixels of the satellite image we predicted and the pixels of the ground truth satellite image. By dividing all the square values we collect by the total number of pixels, we find the mean square error. Then we divide this square error we found by the peak value we found at first. In this way, we calculate the value of PSNR (Peak Signal to Noise Ratio). The higher the PSNR (Peak Signal to Noise Ratio) value, the more efficient and well-formed the new satellite image we created. As a result of our experiments, we stated the result of the best model in the Experimental Results section.

3.4. Hyperparameters

Our first parameter is the k value, which indicates how many satellite views we will choose to reduce the cloud cover effect. The situation that we had to pay attention to when determining this k value was this. When we deleted satellite images with a cloud cover value of 15% or more, the number of satellite images we had should have been equal to or greater than k. For this reason, we have determined 7 as the most optimal k value. The value of batch you in the training phase of model was a problem for us. Although we normally think that the higher batch size value are more optimal, we determined the batch size value 8 due to the hardware shortage. When Batch size is 8 or more, we can't keep all of the satellite data in RAM. We set a high number of epoch depending on the learning rate value. The number of epoch we set is 100. The learning rate value, which optimizes the Loss value as a result of the values we make, is 0.0000005. Due to the epoch and learning rate values we have specified, even though the train time is longer, the loss value is the most optimal decrease value. The results we obtained as a result of these parameters are shown in the Experimental Results section.

4. Experimental Results

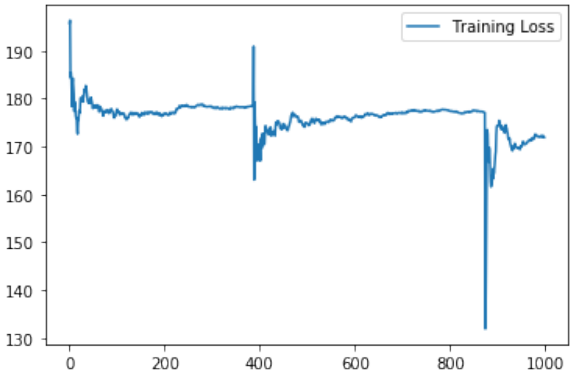

Figure 7. Train Loss

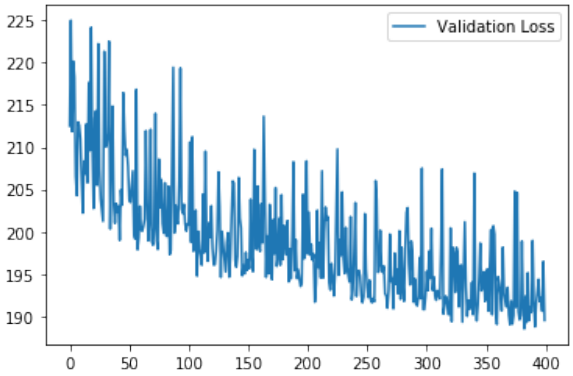

Figure 8. Validation Loss

PSNR (Peak Signal to Noise Ratio) Result = 48.880

Here we see a high resolution patch figure 9 created from figure 10 and 6 more of the low resolution scenery. We see the edges are sharp and crisp, the overall structure is preserved and details not on the original low resolution image is gained by ther low resolution frame and represented in the final output of the work.

Here we see the model failing to produce any meaningful good upscaled patch.We also observe a high non-valid pixel count on the original HR image. We also see a low-resolution image for this scenery where the shift amount is

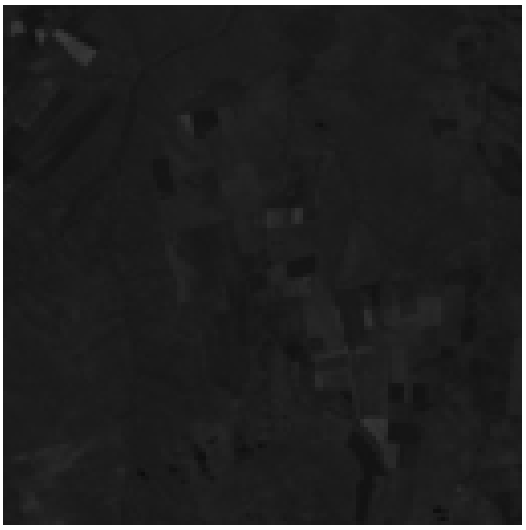

Figure 9. Low Resolution 2

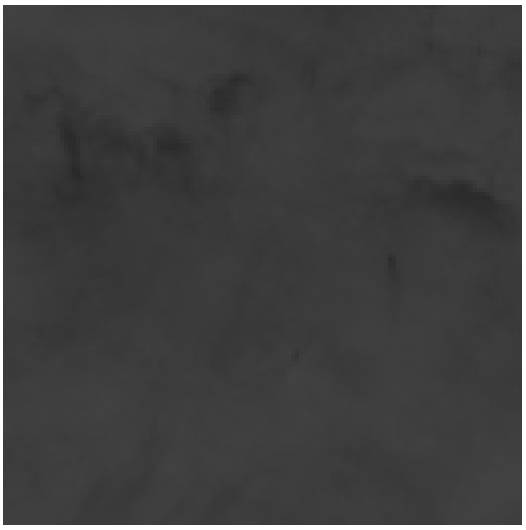

Figure 11. Low Resolution 1

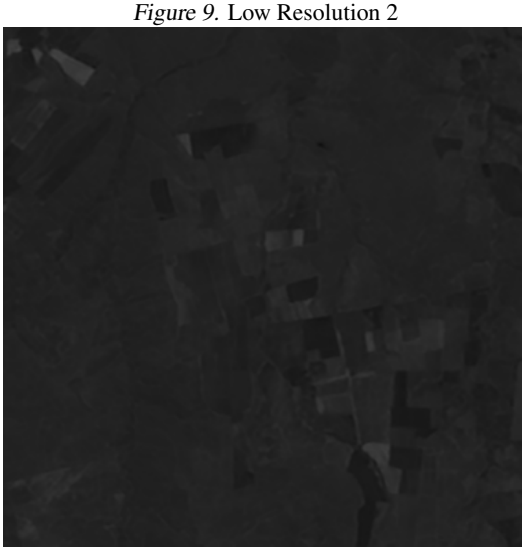

Figure 10. Our Prediction 2

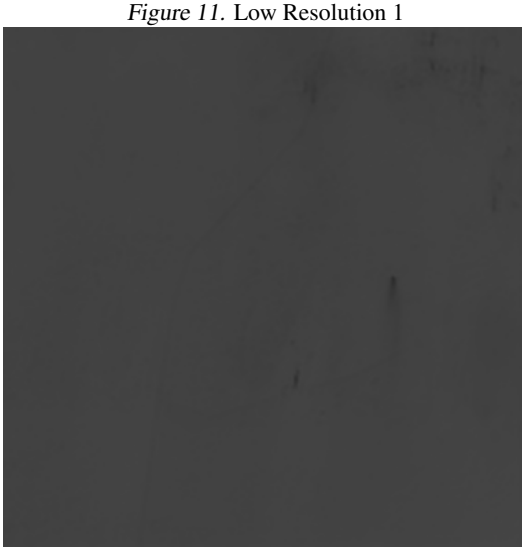

Figure 12. Our Prediction 1

above the expected value of 6 in the upward direction. This produces a completely mis-aligned shape but we still see that values that do not change much in the x gradient are kept the same such as the smudge on the right side of the figure 12.

5. Discussion and Conclusions

In our work we've seen the information not present in any single low resolution image to be able to gathered by the network over the whole image batch presented. 3D convolutions are able to capture the details and the specific wdsr-b blocks to be able to map the relationship really well. A relatively low resource model of 0.5 M parameters is enough to produce really meaningful upscale operation over the original image batches. This work we hope to be able to be used in creation of high resolution images from the available public low to mid resolution satellite imagery. The huge reserve of satellite imagery will be leveraged much better for the scientific work without huge costs of satellite imagery retrieval from private companies. We present this work as out project submission for BBM416 taught by Nazlı İkizler-Cinbiş, we would like to thank our instructor and ESA for the dataset provided also the Google Colab platform used in this project.

References

[1] Al-Wassai, Firouz A., and N. V. Kalyankar. "Major limitations of satellite images." arXiv preprint arXiv:1307.2434 (2013).

[2] Cheng, Chio-Zong Frank, et al. "Satellite constellation monitors global and space weather." Eos, Transactions American Geophysical Union 87.17 (2006): 166-166.

[3] Taini, Giacomo, Andrea Pietropaolo, and Anna Notarantonio. "Criteria and trade-offs for LEO orbit design." 2008 IEEE Aerospace Conference. IEEE, 2008.

[4] Irons, James R., John L. Dwyer, and Julia A. Barsi. "The next Landsat satellite: The Landsat data continuity mission." Remote Sensing of Environment 122 (2012): 11-21.

[5] Savtchenko, A., et al. "Terra and Aqua MODIS products available from NASA GES DAAC." Advances in Space Research 34.4 (2004): 710-714.

[6] Zhu, Xiaolin, et al. "Spatiotemporal fusion of multisource remote sensing data: literature survey, taxonomy, principles, applications, and future directions." Remote Sensing 10.4 (2018): 527.

[7] Dong, Chao, et al. "Image super-resolution using deep convolutional networks." IEEE transactions on pattern analysis and machine intelligence 38.2 (2015): 295-307.

[8] Ledig, Christian, et al. "Photo-realistic single image super-resolution using a generative adversarial network." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[9] Yang, Jianchao, et al. "Image super-resolution via sparse representation." IEEE transactions on image processing 19.11 (2010): 2861-2873.

[10] Xu, Bing, et al. "Empirical evaluation of rectified activations in convolutional network." arXiv preprint arXiv:1505.00853 (2015).

[11] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009.

[12] Shi, Wenzhe, et al. "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network." Proceedings of the IEEE conference on computer vision and pattern recognition.