Slum Mapping in Islamabad using Satellite Images

Hadi Askari

Abstract

The traditional method of finding slums are manual surveys by physically visiting areas or by manual inspection of imagery. In this work, we propose an automatic method of finding slums from satellite images. High-resolution satellite images, like Sentinel-2, are available globally and free of cost. We propose a Convolutional Neural Network (CNN) based method of automatic recognition of slums from overhead images. We prepare a dataset for training and evaluation of automatic detection of slums from overhead imagery. Once trained, our method can be used on freely available satellite images of new areas. This can be extremely helpful in targeting already poverty stricken areas thus helping in the allocation of resources and better city planning.

1. Dataset

To create the dataset, the initial step was downloading recent Sentinel-2 satellite images of Islamabad,Pakistan. These images were multi spectral so I had to learn how to extract the RGB bands and display them in python. Sentinel-2 images are geoTiff images so they proved challenging to work on with traditional image processing libraries like OpenCV and PIL. So I had to learn other specific libraries like Rasterio and Earthpy. In total, we collected 1692 tiles, each covering around 3.6km x 1.93km area with 10-meter resolution per pixel. We reserved 1184 tiles for training, 254 tiles for validation, and 254 tiles for testing.

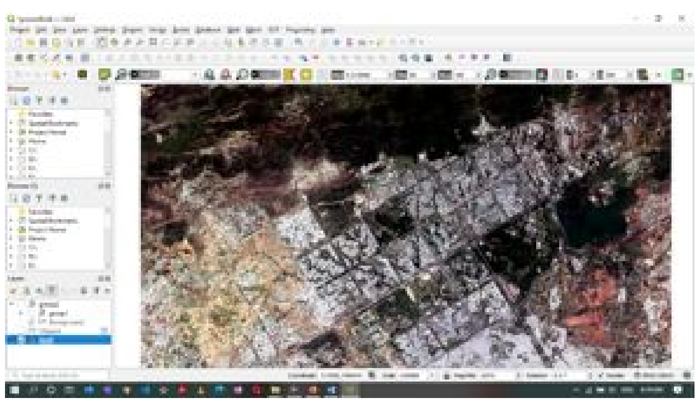

Figure 1. Original Sentinel-2 rastershown in QGIS.

Next, I learnt how to use the software QGIS and used it to construct the ground truth images for training. I hand drew polygons on the Sentinel-2 map of Islamabad based on known slum areas. These masks were then superimposed on the actual satellite images in python. Finally, tiling of these images was done by making grids of 64x64 pixels and these constituted the dataset for training.

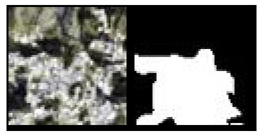

Figure 2. A sample image and its mask showing the slum region.

2. Model and Training

We formulate the problem as a per-pixel classification task, commonly known as semantic segmentation. The semantic segmentation network used was UNet [1] with Resnet34 [2] as the backbone architecture. Other semantic segmentation algorithms like FCN [3] could also have been used. UNet was primarily developed to process biomedical images and it usually performs better for remote sensing tasks. Since we have two classes, we modify UNet to have two output channels and use Softmax activation on the final layer.

We train our network with backpropagation using the cross-entropy loss, given as:

$$-(y\log(p) + (1-y)\log(1-p))$$

where p is the model prediction and y is the true label. The model was trained for 35 epochs using weight decay of 0.01 as regularization and the learning rate (lr) was $8\text{x}10^{-5}$ sliced from lr/400 to lr/4.

3. Evaluation

For quantitative evaluation, we use the commonly used metric of DICE score which is given by:

$$Dice = \frac{2 \times TP}{(TP + FP) + (TP + FN)}$$

where TP is True Positive, FP is False Positive, and FN is False Negative. Our model achieved the maximum **DICE** score of 82.5%. Our results of the final epoch are given in Table1.

Epoch	**Training Loss**	**Validation Loss**	**DICE**
33	0.074277	0.080042	0.825105

Table 1. Quantitative results of our model on the held-out test set.

Our qualitative results are shown in Figure 3. We can see that model predictions closely align with the true labels of slums.

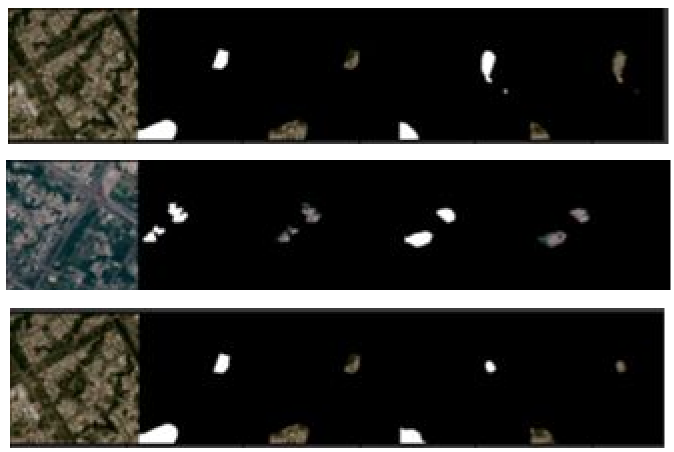

Figure 3. Qualitative results of our model. From left to right: image, ground truth, and prediction.

4. Challenges

In this project, I learnt the whole process of deep learning, including dataset labeling, preparing data loaders, and using the modern deep learning framework PyTorch for end-to-end optimization. In this process I learned the significance of having mutually exclusive training and test sets to make sure that the model generalizes to unseen data. The biggest problem related to this data was manipulating the bands and displaying the RGB image of these multi-spectrum images on python. I eventually had to use histogram equalization to make the stacked bands visible, even though they still had a slight tinge to them. Mastering QGIS also took some time as I wasn't accustomed to its interface. Class Imbalance was another issue since Islamabad is a planned city that doesn’t have a lot of slums. By preparing a roughly balanced training dataset, we were able to avoid bias in the network that usually arises due to class imbalance.

5. Future Possibilities

Full training of deep CNNs requires a large number of labeled examples for training. Since training data is limited in our case, a standard solution will be to use transfer learning for future slum mapping in different cities. Other semantic segmentation algorithms (such as the ones mentioned above) can be used to compare the results. Another solution is to use few-shot learning and see if it can be utilized to further improve results and increase computational efficiency.

Conclusion

This was a highly engaging project encompassing all of the steps in a typical research oriented deep learning project. The data set generation was challenging and broadened my understanding of how to manipulate images for semantic segmentation. I also enhanced my knowledge about various deep learning algorithms, their applications and implementations while researching for this project.

Acknowledgements

The project was supervised by my faculty advisor Dr. Muhammad Shahzad.

References

[1] Olaf Ronneberger, Philipp Fischer, Thomas Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, “Deep Residual Learning for Image Recognition,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[3] Jonathan Long, Evan Shelhamer, Trevor Darrell, “Fully Convolutional Networks for Semantic Segmentation,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.