Road Detection in Satellite Images

Intermediate Report

Aradhya Neeraj Mathur
PHD17008

Priyabrata Mallick
MT17043

Viraj Parimi
2015068

***Abstract*—The main problem that we aim at working is the detection of roads in satellite images. The accurate detection of roads is imperative to any navigation. Good detection can help create better guidance systems for the vehicles. We will look into different methods that can help get better detection of roads in such satellite images. It is important for a mapping system to be adept and robust to different locations and subtleties of construction.**

I. Introduction

It is very difficult to maintain accurate and updated maps, as it is very tedious to create and expensive to maintain. Multiple companies have invested large amounts, still there is a gap in accuracy. It is very difficult to extract roads from aerial images due to the occlusion by trees, buildings and shadows. Our project aims at exploring this domain through different techniques.

II. Dataset

There are two datasets that we use in the project. We use the dataset https://www.cs.toronto.edu/ vmnih/data/ which contains the satellite images and the corresponding road masks.

The second dataset that we use the MIT roadtracer dataset. The roadtracer dataset is a large dataset consisting of high resolution satellite imagery along with the corresponding road map masks. The data was generated from a large dataset of *planet.osm.pbf* which contains openstreetmap of size 47GB. From these maps of different regions are extracted. There are 40 regions in the dataset. It is further divided into a train and test set of 25 and 15. Each image is of dimension 4096×4096. Since images and large and high resolution we check whether we can reduce the size of the images and their corresponding road masks. Since the images can be rescaled without loosing the correspondence by checking the images of roads obtained after rescale and mask multiplication; we rescale the images and generates patches of size

III. Literature Review

As part of the literature review that we conducted we divided it into 2 main parts. The works that have been conducted on segmentation in general and the works that have been conducted on road segmentation in specific.

Identify applicable funding agency here. If none, delete this.

Fig. 1. a) Satellite Image b) Mask Image

Segmentation based approaches first segment an image then classify the regions instead of each pixels. Kettig and Landgrebe [Kettig and Landgrebe, 1976] proposed a procedure that first classified aerial image patches of size 4 by 4 into homogeneous and non-homogeneous regions. The non-homogeneous regions were then classified pixel-by-pixel, and the homogeneous regions were merged into larger regions and then classified using a region classifier. We looked at different techniques that have been used previously for segmentation starting from classical approaches such as Otsu's thresholding method for segmentation [1], seeded segmentation [2] to machine learning methods such as K-Means for segmentation and more advanced state of the art deep learning methods such as U-Net for Segmentation.

Another method for road extraction from satellite images is explored in [3] where instead of directly using segmentation he authors propose a decision function implemented in the form of a CNN that computes the road graph iteratively. The search begins from a location on the road then the vertices and edges are added depending on the decision from the CNN. An input region (v_0, B) is given to the algorithm. From this starting location the algorithm maintains a graph G and a stack of vertices S that initially contains v_0. Then the decision function decides either to walk a fixed distance along a certain direction or stop. The direction is determined by selecting angles that are uniformly distributed in $[0, 2\pi)$.

IV. Experiments

The pipeline of the project is as follows:

- Preprocessing
- Segmentation

- Generating the graph network

We will now present a step-by-step process wherein we will tackle each problem independently and finally will try to piece together each part of the puzzle.

A. Preprocessing

For preprocessing, we first of all subsampled each image and their corresponding ground truth by a factor of 4. This step was necessary as each original image was of the size of 4096*4096 which would have significantly impacted on the computation time of any method that was to follow. In order to justify that subsampling of both the image coupled with the ground truth would not cause much distortion, we performed an independent experiment wherein we utilized the Hadamard product and plotted the new image. Later we compared this image with the original image with respect to roads and found that there were no significant distortions thereby validating the subsampling process. The same can be seen in the following figures.

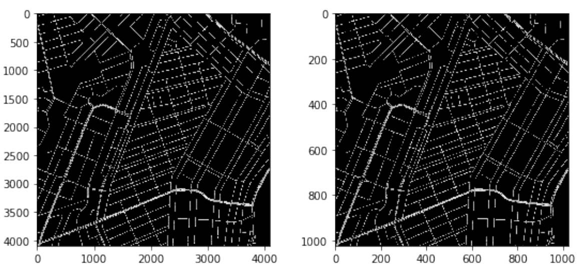

Fig. 2. a) Original Image Masked b) Downsampled Image Masked

Apart from this we also emperically compared the histograms of both the above images using Bhattacharya distance measure which indicates that lower the value (0-1 range), closer are two histograms. Based on this metric we found that the imaged were 0.11 distance apart thereby showing that both the images are pretty similar and wont cause much issues in later stages of the pipeline.

B. Segementation

The next step in the pipeline is segmentation. As we have studied multiple methods in the class with regards to this topic, we first tried to use those concepts in our project.

- Kmeans - Given that there are only two classes in our dataset, one signifying *road* and another signifying *not road*, we performed K-means clustering by setting K = 2. We found it did not work effectively as it unable to distinguish the roofs and roads properly, similarly forest and river going into the same bucket.
- Otsu - As a naive approach we decided to perform otsu thresholding. After applying this method over several images we found that this method was only effective in some cases. The reason that it did not work in most of the cases was that most of the aerial road images had shadows that were casted on the roads by high-rise buildings thereby changing the color of those pixels. This caused the algorithm to classify those pixels as not part of the road. This is evident in the following images

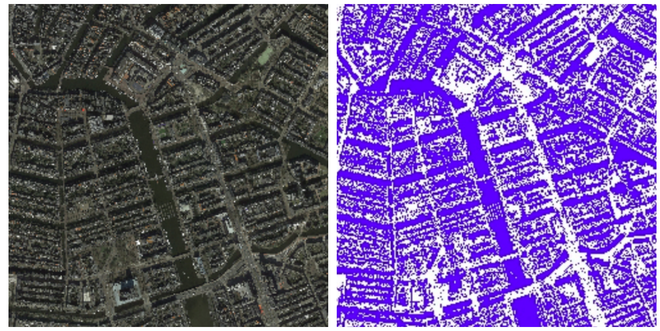

Fig. 3. a) Aerial Image view b) Clustering using K Means(K=2)

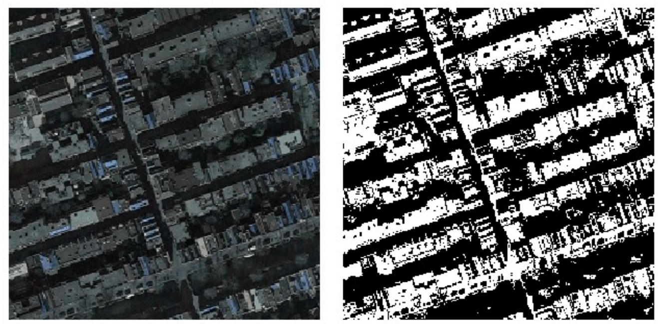

Fig. 4. a) Aerial Image view b) Output of Otsu Thresholding

- Seeded Segmentation - Given that the first two methods did not turn out well, we decided to give seeded segmentation a try. This method performs well if the seed pixel was chosen correctly and even after that in most cases we observed that the algorithm would only process in one direction of the neighborhood pixels meaning that it would leave the other end of the road unmarked thereby making this method unfavourable as well. This deficiency is visible in the output images shown below.

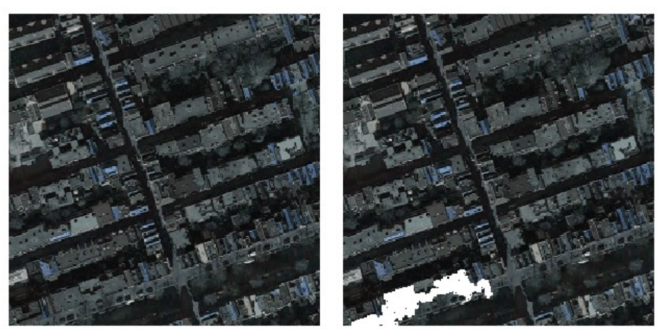

Fig. 5. a) Aerial Image view b) Output of Seeded Segmentation

- U-Net - Given that none of the conventional Computer Vision based segmentation approaches worked upto our expectations, we decided to move to deep-learning architectures. We did more literature survey and found that most people use U-Net or Mask-RCNN for segmentation purposes. But since we were constrained with respect to computation resources, we decided to go with U-Net. To perform the training we first did more preprocessing to the images. Firstly, we divided a training image into patches. Secondly, we considered only those patches which had atleast 100 pixels corresponding to roads in the ground truth. Finally, we trained U-net for segmenting the roads given the patches and their corresponding ground truths. The currently obtained IoU is **0.53** given the imbalance of class since it performs a per-pixel classification.

C. Graph

To perform this task we decided on a naive approach whose pipeline is as follows

- Keypoint extraction
- Network formation

To summarise, the first step would be to extract the keypoints in the image such as corners that could potentially depict the intersections and then pass these keypoints to the next step wherein we would decide to add an edge between a pair of keypoints if and only if there is a road connecting between them in the segmented output. Currently we are only at the first step and have not been able to perform the second step.

1) Keypoint Extraction: For keypoint extraction we perform erosion using a 3×3 filter and then perform corner detection to obtain junction points and intersections. Erosion is necessary to reduce the width of the road to get better corners and prevent spurious detections. Apart from this technique we also performed Harris corner detection algorithm. The results of these approaches are shown in the following images.

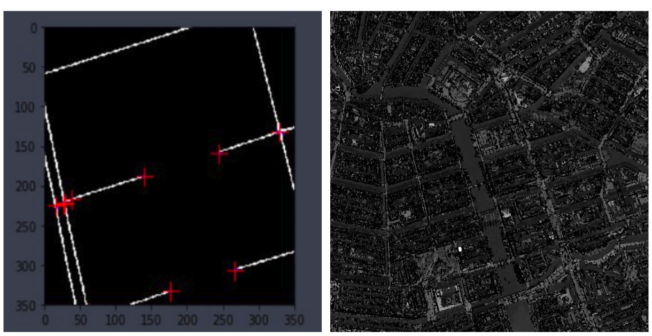

Fig. 6. a) Erosion b) Harris Corner Detector

V. Future Work

- We plan on training our U-Net model for more epochs and use other metrics such as dice-coefficient and ROC plots to perform more in-depth analysis of the results that we have obtained.
- We also plan on proposing a more refined approach towards graph creation

References

[1] Vala, Hetal J., and Astha Baxi. *A review on Otsu image segmentation algorithm.* International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) 2.2 (2013): 387-389.

[2] Shih, Frank Y., and Shouxian Cheng. *"Automatic seeded region growing for color image segmentation.* [*Image and vision computing 23.10 (2005).*]. 877-886

[3] Bastani, Favyen and He, Songtao and Abbar, Sofiane and Alizadeh, Mohammad and Balakrishnan, Hari and Chawla, Sanjay and Madden, Sam and DeWitt, David *Roadtracer: Automatic extraction of road networks from aerial images*. [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*].

[4] R. L. Kettig and D. A. Landgrebe. *Classification of multispectral image data by extraction and classification of homogeneous objects.* IEEE Transactions on Geo- science Electronics, 14:1926, 1976.

[5] Volodymyr Mnih *Machine Learning for Aerial Image Labeling* PhD Thesis

[6] GitHub repository. *https://github.com/viraj96/CV*$_{Project}$