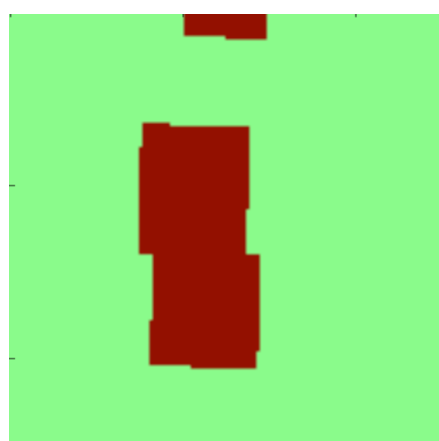
**Satellite Image Processing Learning Algorithm**
EECS 349 Northwestern University Spring 2016

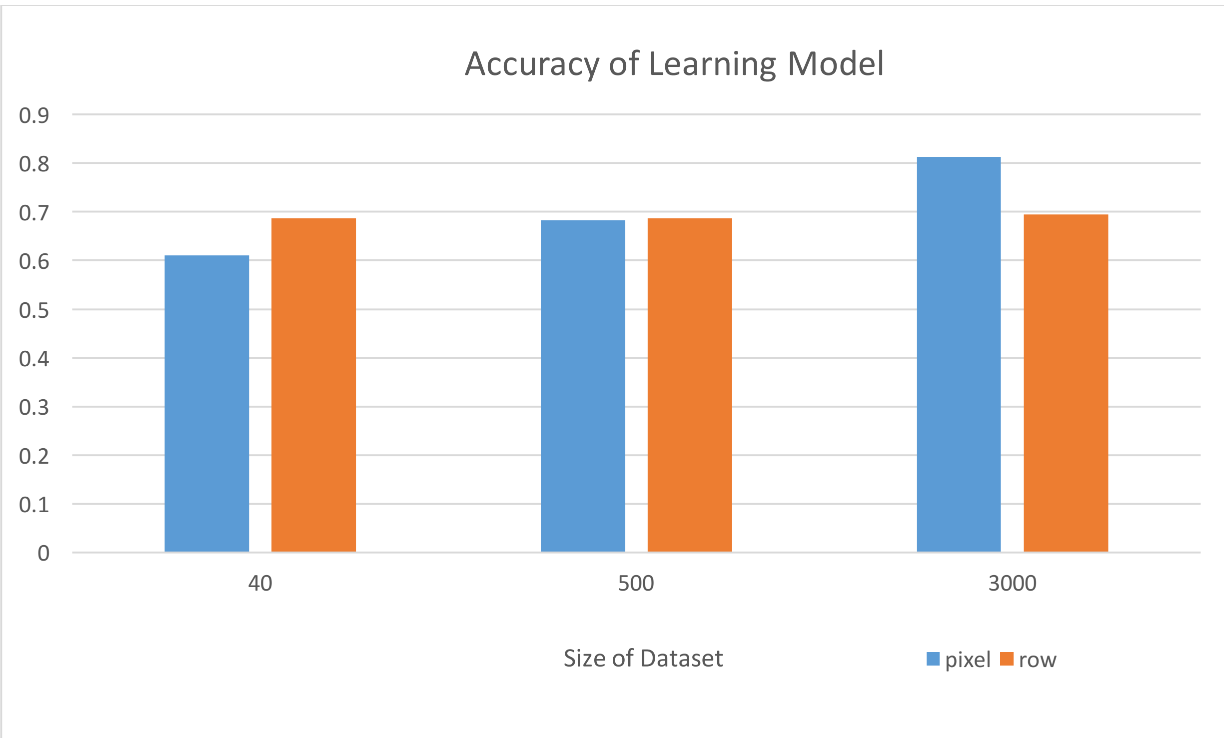
Nikita Patel and Nikhil Charles
nikipatel@u.northwestern.edu | nikhilcharles2017@u.northwestern.edu

This project uses machine learning for satellite imagery processing to identify building rooftops. The results of this model can be applied in many situations, and would be especially useful in identifying residential areas in under documented regions of the world. For example, suppose there is an Ebola outbreak in Liberia, and the WHO wants to go in and help mitigate the problem by quarantining the infected and properly disposing of the people who passed away from this disease. They know they can start in the main cities, but they are unsure as to how far the epidemic has spread to small towns and villages. In addition, they don't know exactly where households in rural Liberia are located. This model will allow them to quickly identify where they should be looking and the path that the Ebola outbreak may have taken based on the proximity of houses and villages. Rapid image processing will prove to be very useful in the near future as the image capturing technology improves, becomes higher quality, and includes more meta data.

The data set that is being used for this project as the training and testing data is from Mass GIS. It includes a jp2 satellite image of Amherst, Massachusetts as well as a shape file that contains a digitized model of all the rooftops in the state of Massachusetts. We converted both jp2 and shape files into tif files. We then segmented these tif files each into 50,000 equal sized tiles to use as training and test examples. We will call the tiles of satellite imagery 'image tiles' and the tiles of digitized rooftops 'label tiles'. To use this data set of tif files, we then extracted the meta data from both the image and label tiles so that they were matrices of data. The image tiles contain the RGB values as well as an infrared value for each pixel in the matrix. Each pixel in the label tile contains a 1 if the pixel corresponds to a rooftop and a 0 if it does not. The data then had to be processed in a specific way for each method the was tried over the course of this project. The images below show an example of an image tile and label tile pair.



This project uses a structured learning model called Chain Conditional Random Fields (Chain-CRFs). This model uses contextual information beyond the features of one case being examined in order to classify the instance. In the case of this project, the model takes in the R, G, B, and infrared values for each pixel in an image, and uses these features alongside the other pixels in the image to predict which pixels are part of a rooftop and which are not part of a rooftop. A library called PyStruct was used to implement this learning algorithm. The dataset was used with this structured learning algorithm in two ways, as follows. The first inputted the dataset as is, assigning a classification value to each pixel in the image. The second inputted the dataset with assigned classifications per row – so each row is labeled for whether or not there is part of a building is in it. Below are the accuracy results from using three different dataset sizes for each of the two ways to label data. Each of these datasets were split as 66% training and 34% validation.




As expected, the accuracy of both pixel and row classification strategies increased as the dataset size increased. However, what was interesting is that the pixel classification performed much better than the row classification in the largest dataset as well as increased by a larger margin across the three dataset sizes. In addition, the margin between pixel and row classification for the 3000 file dataset was much larger than the other two dataset sizes. When looking at the actual accuracies, the row classification remains just under 70%. This shows that while this is a slightly successful method, it doesn't improve as it's dataset improves and is not really able to learn much from the data. The pixel classification however improves greatly with the increase in dataset size. In addition, it had a final accuracy of 81% with the 3000 image dataset.

The next steps for to improve the accuracy of this model further would be to test with even larger datasets. Based on the current results, it would seem that the row classification method would further improve the accuracy. It would also be useful to implement and test different structured learning methods that the PyStruct library supports as well as other libraries like CRF++ and seqlearn.

Division of work:
Niki – initial research on image processing and structured learning, learning model and PyStruct implementation, abstract and final report
Nikhil – learning model and PyStruct implementation, dataset manipulation and classification method testing, website

**References**


http://web.mit.edu/profit/PDFS/EdwardTolson.pdf

http://www.vision-systems.com/articles/print/volume-20/issue-2/features/machine-learning-leverages-image-classification-techniques.html

https://pystruct.github.io/

https://pystruct.github.io/auto_examples/plot_letters.html#sphx-glr-auto-examples-plot-letters-py

http://maps.massgis.state.ma.us/map_ol/oliver.php