EE/CprE/SE 491 WEEKLY REPORT XY

11/24/2019 – 12/01/2019

Group number: 20-18

Project title: Development of Image Analysis Algorithms for Crack Detection Using a Smartphone

Client &/Advisor: Bo Yang/Halil Ceylan

Team Members/Role: Akira Demoss, Maggie Dalton, Modeste Kenne, Nik Thota

---

(All the above information should be there in each weekly report. The format/color scheme etc need not be the same. However, please remove everything that is in a bracket from your final submission. These are just part of the template and need not be a part of the report.)
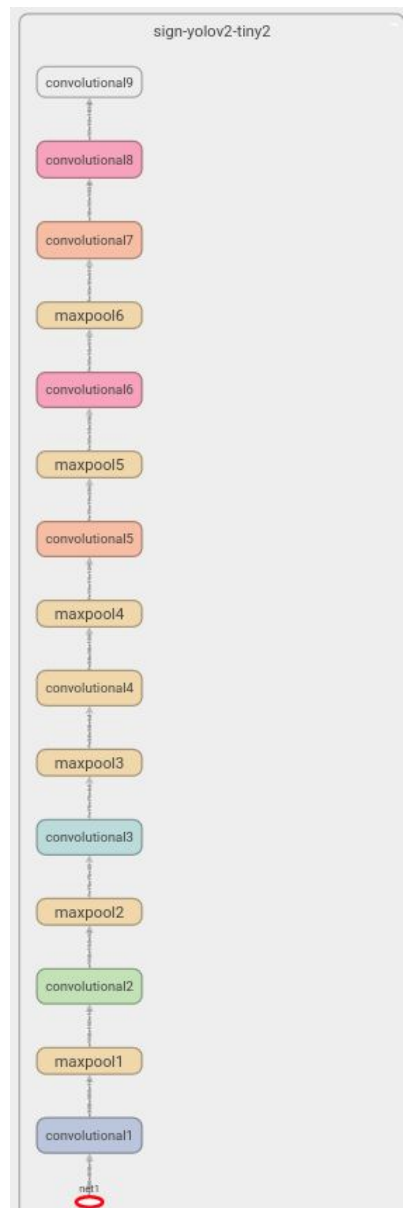
- **Weekly Summary**
    - This week we have spent tying up loose ends for our projects and preparing demos of two components of our final application.

- **Past week accomplishments**
    - Maggie
        - Set up and ran the trained model for detecting cracks on videos as a simulation for how the algorithm will perform while driving
            - Recorded some videos of driving down streets in Ames with cracks for testing
            - Found additional videos from YouTube of driving in Iowa City and Des Moines
    - Akira
        - Standardized a way for reporting on issues with setting up the training environment using a standard Issues Template.
        - Resolved issues 1 from the team repository. This error involved discerning

between the correct command line arguments to use which would allow android to recognize the model.

- ○ Resolved issue number 2. This involved figuring out and understanding how the tensorflow frozen graph was converted to a tflite model. Specifically I learned that the command line argument that requires the input shape of the frozen graph determines the number of bytes that the file will be output as. Hence if shape=(1,300,300,3) then the output will be 108,0000 bytes when we are using a data type that is equal to float32 (4 bytes). Hence 1*300*300*3*4 gives us the desired size to input into java which is 108,000.
- ○ Created a detailed tutorial on how to train a custom neural network with darknet and how to understand the output that is displayed in the terminal while training
- ○ Also created documentation detailing how to analyze the generated tensorflow graph using tensorboard. This generates a url on localhost which will allow you to view the node names of your tensorflow graph in an interactive GUI. Below is the architecture of our neural network.
- ○ Installed and created a tutorial documenting all steps on how to install Tensorflow with GPU support form sources.
- ○

sign-yolov2-tiny2

convolutional9

convolutional8

convolutional7

maxpool6

convolutional6

maxpool5

convolutional5

maxpool4

convolutional4

maxpool3

convolutional3

maxpool2

convolutional2

maxpool1

convolutional1

net1

- Modeste
  - Learned how to use node.js to send files on server
  - Set up node.js modules and require dependencies necessary to compile its native modules (Chocolatey/visual studio-workload-vctools)
  - Set up a MongoDB. Also tried using MySQL to run a couple of Node.js experiments, however I think the latter is more suitable for our project because of its scalability.
  - Created a couple of Android Studio projects to implement Node.js
- Nik
  - Created a bash script to find duplicates in data sets
    - Reviewed bash scripts and how they work
    - Determined how to compare 2 separate file directories
  - Fixed an error with environment setup, which led to a new error

- **Pending issues**
  - Akira
    - Many of the issues I am resolving are obscure, as I am still in the process of familiarizing myself with the darknet framework and running all of the necessary conversions to make this run on Android. Other algorithms and frameworks have more support, but considering the requirements of the project we believe that sticking with the initial plan of using yolo with TensorflowLite will yield the fastest and most accurate results, hence persevering through these challenges is a necessary part of creating a realistic application.
  - Maggie
    - While the algorithm seems to perform extremely well on images it does not perform very strongly on videos. It will rarely, if ever, detect a crack in a video. Currently, we are unsure of the root cause (whether it's angle/lighting/speed/resolution/need for more diverse training data/etc.). Have tried changing the threshold/confidence levels for displaying a detection but it appears that outside of the existing displayed detections, detections may not be being made.
  - Modeste
    - After I start the server, I'm able to load an image to to the app. But it fails to send it back to the server. I'm thinking it has something to do with the way I'm using the url and the listening port.
  - Nik
    - When running the bash script, it deletes the last 3 files in the dataset

every time. This is an issue because our dataset will slowly deplete itself everytime we run a duplicate check. I am currently surfing the web for answers on why this is happening.
- ○ I am also still having issues with my environment. I solved the most recent error, but have encountered a new one. I will continue looking into it.
- o **Individual contributions**

Name	Individual Contributions (Quick list of contributions, this should be short)	Hours this week	Hours Cumulative
Akira Demoss	Resolved issues related to performing object detection on Android, documented steps for analyzing a tensorflow graph of our trained detector, trained multiple custom road damage detectors that detect both longitudinal and transverse cracks.	36	118
Maggie Dalton	Ran the crack detection algorithm on videos.	12	96
Modeste Kenne	Improved methodology for client/server communication with Node.js.	30	76
Nik Thota	Created a bash script to delete duplicates in our datasets and labels	30	80

- o **Comments and extended discussion**

- o **Plans for the upcoming week**
  - ● Akira
    - ○ Focus on the presentation and reflecting on the work we have accomplished this semester as well as wrapping up preliminary tests.
  - ● Maggie
    - ○ Gather more videos/images of Iowa roads
  - ● Modeste
    - ○ Debug the app and finalize the client/server communication.
    - ○ Plan on how to tackle next semester's goal
  - ● Nik
    - ○ Debug bash script

- ○ Iron out any issues with the environment and ensure it is ready for use in the spring semester
- ○ Finish presentation and practice it

- o **Summary of weekly advisor meeting**
  - ● We will meet with our advisor later in the week.