Project Report: Detecting Street Signs in Videos in a Robot Swarm

RAHUL RAGHATATE[1,*] **AND SNEHAL CHEMBURKAR**[1]

[1] *School of Informatics and Computing, Bloomington, IN 47408, U.S.A.*
[*] *Corresponding authors: rraghata@iu.edu, snehchem@iu.edu*

S17-IR-P003, March 19, 2017

The aim of this project is to deploy a software package to detect different street signs in a video stream. This will be a scalable system over Hadoop based cloud ecosystem to incorporate multiple video feeds and parallel real-time processing of the feeds. A comparative benchmark will be developed based on the performance of package on multiple cloud systems.

Keywords: Street Signs, Video Streams, OpenCV, Spark, Cloud

https://github.com/rahulraghatate/sp17-i524/project/S17-IR-P003/report/report.pdf

INTRODUCTION

Detecting objects in images has always been keen area of interest in the field of computer vision. There are many applications developed based on this simple idea like auto tagging pictures (e.g. Facebook, Phototime), counting the number of people in a street(e.g. Placemeter), classifying pictures, detecting vehicles, etc.On the similar grounds, we are building a software package which can be deployed easily on cloud infrastructure and establish a platform to detect different street signs in a video stream. A benchmark will be developed based on performance of this software on different cloud systems. The database of street signs will be restricted to US street signs. The video streams used for this project are simulated or captured using mobile camera.

REQUIREMENT ANALYSIS

We are using following technologies for complete project development and deployment:

1. Cloudmesh - For connecting to different cloud environments.
2. Ansible -For deploying software from master node on virtual slave nodes.
3. Python - Writing script for data analysis and data processing in Spark [1] engine over Hadoop.
4. Hadoop -Required for uploading our data set of images on distributed data storage platform as well as for video streams and its processing in Spark Stream. Using pre-build Hadoop and Spark in Cloudmesh so as to focus on video data distribution and analysis and perform optimization testing on cluster.
5. OpenCV [2] -Perform video analysis for street sign detection using open source computer vision libraries of video/image analysis algorithms.The OpenCV library provides several features to manipulate images(apply filters, transformation), detect and recognize objects in images.

METHODOLOGY

1. Data gathering for street signs and video streams
2. Deploy Hadoop clusters on cloud using Cloudmesh
3. Develop Ansible script to install OpenCV on cloud
4. Build a model to detect or track street signs using OpenCV. We plan on implementing two programs-
   - Read an image and run the Haar cascade classifier to detect the signs in the image and
   - use the video stream and detect signs in real time.
5. To detect street signs, we will be using Haar based cascade classifier which detect objects in an image. As detecting signs and categorizing them are two different problems and use two different approaches. Hence, we will benchmark detection first and will work on categorization as future development.
6. Test the performance of software package on 3 different clouds or on the same cluster with multiple nodes.
7. Create benchmarks based on the above results

EXECUTION SUMMARY

This section specifies the week by week timeline for project completion.

1. Mar 6 - Mar 12, 2017 Create virtual machines on Chameleon cloud using Cloudmesh and submit the project proposal.

2. Mar 13-Mar 19, 2017 Deploy Hadoop cluster to Chameleon cloud using Cloudmesh and develop Ansible playbook to install the required software packages to the clusters (OpenCV, etc)

3. Mar 20-Mar 26, 2017 Train data to detect or track street signs using OpenCV. Develop Ansible playbook to setup database and connectivity among multiple nodes.

4. Mar 27-Apr 02, 2017 Capture the results of street sign tracking in video streams.

5. Apr 03-Apr 09, 2017 Test on different cloud systems and define benchmarks.

6. Apr 10 - Apr 16, 2017 Create deployable software package in Python.

7. Apr 17-Apr 23, 2017 Write Project Report

USE CASES

1. Street Sign Detection for autonomous vehicles.

2. Analysis of traffic signs in Google Street View to estimate all signs ahead hence, useful in ambulance , fire brigade services, simplest path finder etc.

BENCHMARK

Benchmarks will be created based on the performance of the software in different cloud environments. The initial analysis will be done on a single short video stream and then on video streams distributed across 2 or 3 nodes. The different cloud systems used for the purpose of benchmarking are Chameleon, FutureSystems and JetStream.

ACKNOWLEDGEMENTS

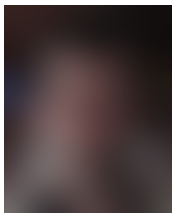
This project is undertaken as part of the I524: Big Data and Open Source Software Projects coursework at Indiana University. We would like to thank our Prof. Gregor von Laszewski, Prof. Gregory Fox and the Associate Instructors for their help and support

REFERENCES

[1] A. S. Foundation, “Overview - spark 2.1.0 documentation,” Web Page, accessed: 03-12-2017. [Online]. Available: http://spark.apache.org/docs/latest/index.html

[2] “Home - opencv/opencv wiki,” Code Repository, accessed: 03-12-2017. [Online]. Available: https://github.com/opencv/opencv/wiki

AUTHOR BIOGRAPHIES



Rahul Raghatate will receive his Masters (Data Science) in 2018 from The Indiana Univeristy Bloomington. His research interests include Big Data and Machine Learning.

Snehal Chemburkar will receive her Masters (Data Science) in 2018 from Indiana University Bloomington. Her research interests also include Big Data and Machine Learning.

WORK BREAKDOWN

Will be updated in later phases of project.

FOLLOWING TOPICS ARE YET TO BE INCLUDED

2.2. Shell Access If applicable comment on how the tool can be used from the command line

3. Licensing Often tools may have different versions, some fre, some for pay. Comment on this. For example while a tool may offer a comercial version this version may be too costly for others. Identify especially the difference between features for free vs commercial tools.

Sometimes you may need to introduce this also in the introduction as there may be a big difference and without the knowledge you do not provide the user an adequate introduction.

4. Ecosystem Some technologies have a large ecosystem developed around them with extensions plugins and other useful tools. Identify if they exists and comment on what they can achieve

provide potentially a mindmap or a figure illustrating how the technology fits in with other technologies if applicable.

5. Educational material Put information here how someone would find out more about the technology. Use important material and do not list hundreds of web pages, be selective.