Real-Time Cyber Physical Systems
Application on MobilityFirst

Winlab Summer Internship 2015
Karthikeyan Ganesan, Wuyang Zhang, Zihong Zheng
Shantanu Ghosh, Avi Cooper
TEAM MEMBERS

Karthikeyan Ganesan  
Wuyang Zhang  
Zihong Zheng  
Shantanu Ghosh  
Avi Cooper
BIG PICTURE OF OUR PROJECT

CPS Application based on MF

MobilityFirst Virtual Network

Server side:
Implement server application for object recognition;
Return the result

Client side:
Run an instance of camera system;
Transmits video in standard format;
Simple graphical interface to display results
Image Recognition:
Implemented the strategy to increase the accuracy and also cut down the mismatch issues.

Application:
Finished the combination of Glass-to-Phone and Phone-to-Node communication. The Google Glass is now able sending image to server and getting back the result.

Cloud Computing:
All configurations are done. The image recognition program could be run in a certain number orbit nodes side-by-side with STORM framework.
Some essential efficiency-related experiments are done.
Some Images From The Google Glass

Index: 19
Successfully Matched!

Index: 4
Successfully Matched!
Background:
As a server processing a large amount of real time requests, requested images will be set into a queue waiting for be processed. They have to keep waiting before the server completes their previous requests, which delays unnecessary requested time.
Spout: Water Source (tweets, images, news...)  
Bolt: Water Processing Machine (popular tweets ranking, image matching, news follower counting)
STORM Application

Processing Time of 1000 Images

- Serial: 365,246 ms
- 2 Bolts: 182,175 ms
- 4 Bolts: 89,825 ms
- 8 Bolts: 47,036 ms
Result Analysis

Curve tendency?
The relation curve between the number of bolts and processing time is roughly a linear method.

Request interval and unit service time?
When the request interval is larger than the unit service time, the actual total service time equals to that of the unit service time but without any waiting time in a queue.

To achieve the service time will not involve the queue waiting time, the serial method allows the minimal request interval is 365ms, but only 47 ms for 8 bolts!!
THE WHOLE CONNECTION

Android phone

Google Glass

Bluetooth

MF

WIFI Access Point on Master Node

Master node as Server manages/allocate image recognition jobs

ORBIT nodes

STORM

Generate

Slaves nodes
Combination of CPS server and STORM cluster

1. Send an image & request result
2. Notify the Spout about the new request
3. Allocate a Bolt to take this request
4. Response the server and get the image
5. Image Recognition
6. Send back the result
7. Return the result
Next Week Plan

Try to improve the stability of both the Google Class and Android phone programs.

Do the combination of our CPS server and the Image Recognition program based on STORM.

And of course the Poster!
Questions?