Real-Time Cyber Physical Systems

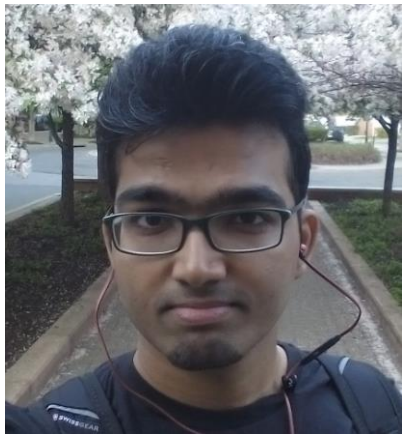
Application on MobilityFirst

Winlab Summer Internship 2015

Karthikeyan Ganesan, Wuyang Zhang, Zihong Zheng

Shantanu Ghosh, Avi Cooper

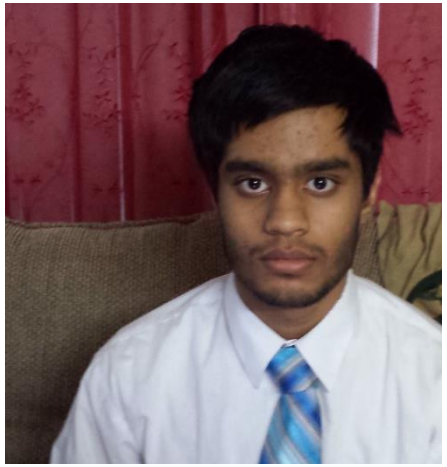
TEAM MEMBERS



Karthikeyan Ganesan

Wuyang Zhang

Zihong Zheng

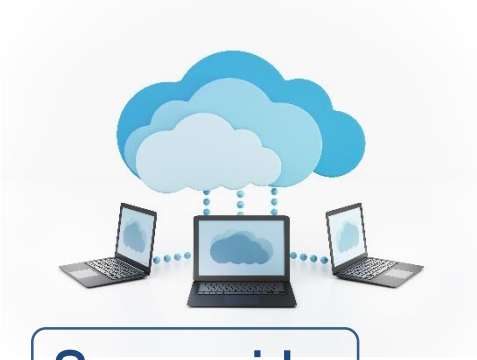


Shantanu Ghosh

Avi Cooper

PRELIMINARY GOAL OF OUR PROJECT

CPS Application based on MF



Server side:

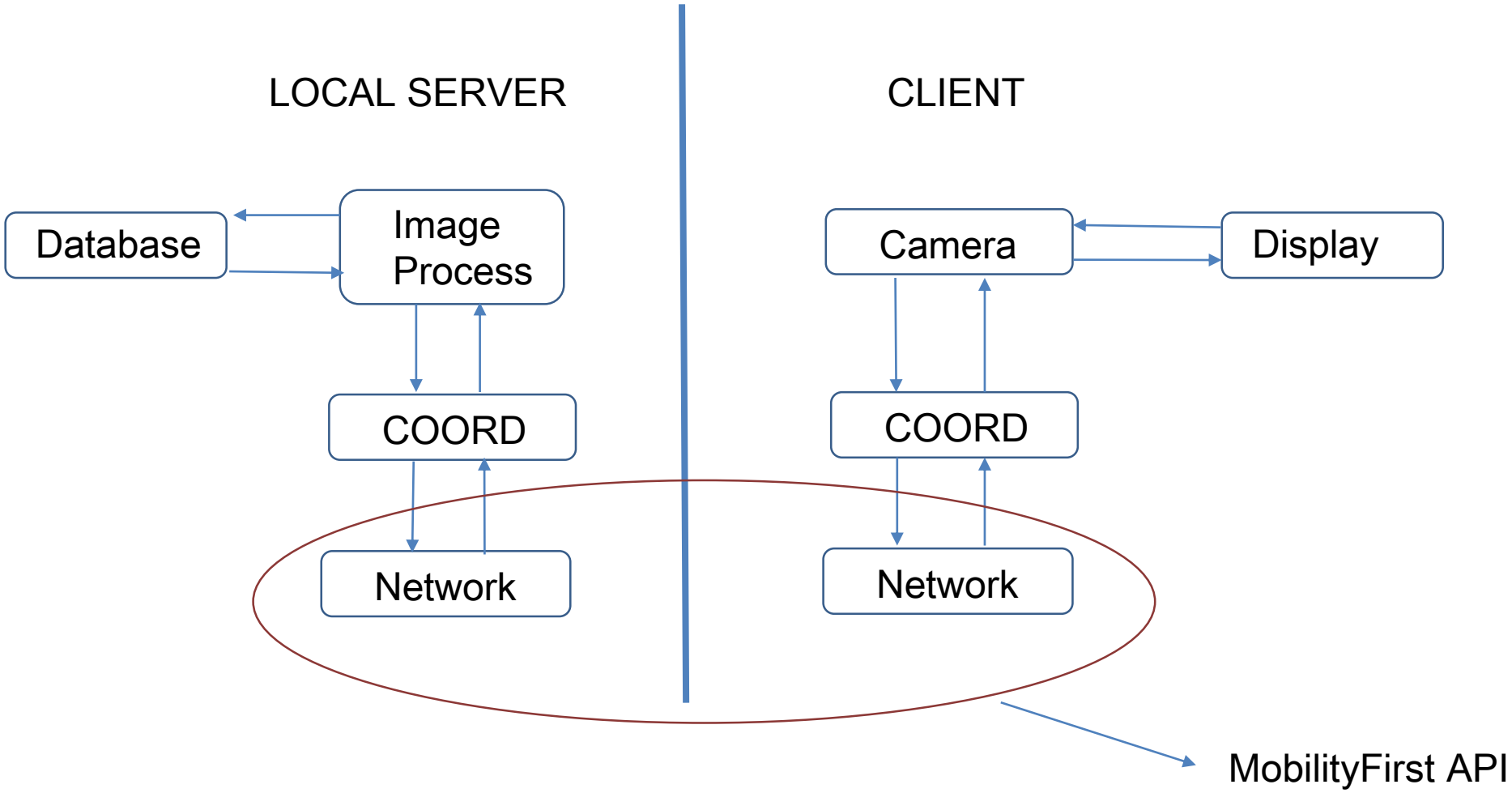
Implement server application for object recognition;
Return the result

MobilityFirst Virtual Network

Client side:

Run an instance of camera system;
Transmits video in standard format;
Simple graphical interface to display results

OUTLINE OF THE PROGRESS



CURRENT FRAME

Application:

Linux APP: Achieved the Museum Scenario function. The Java version client is also built up.

Google Glass: Got the Bluetooth transfer working. Doing combination now.

Cloud Computing:

Successfully set up the Hadoop Environment in ORBIT.

Image Recognition:

Reduced the processing time by removing the tree building phase. Trying to reach a solution using the ratio of matched descriptors to gain greater accuracy.

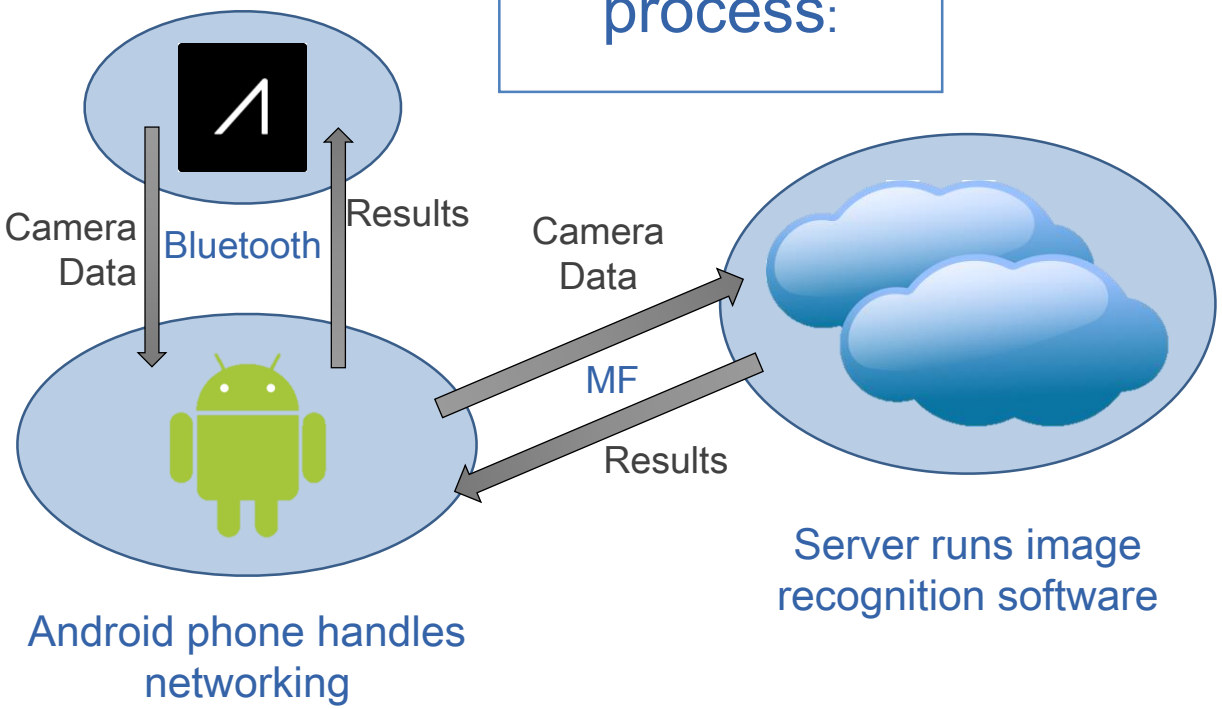
Data Encryption:

Working on OpenSSL stuffs.

Google Glass

Why Google Glass? Glass provides us with access to both a camera that is recording exactly what the user is seeing, and a way to give information back to the user unobtrusively. What the user sees and how they sees it is very important.

Outline of process:



Why the phone as a go between? Glass is relatively low on battery and computing. Transferring networking to a phone will improve both battery life and networking speeds.

Bluetooth transmission

Bluetooth specs: Bluetooth can only send 20 byte packets at a time so my solution was to:

1. Split the byte array of the file (the image from the camera) into 17 byte packets
2. Give each packet a heading of its place among the other packets:
    1. each byte has a range of 256. So with 3 counter- header bytes, times the 17 important bytes in the packet, there is a maximum transmission size of 285,212,672 bytes, or 272 MB. (With only 2 header- bytes, though there are 18 body- bytes, the maximum transmission size will only be 1.168 MB)
3. Receive each packet on the server side and place it into a 2D array, ordered by their headers.
4. Iterate though the array, pulling out only the body- bytes and place them into a single byte array.

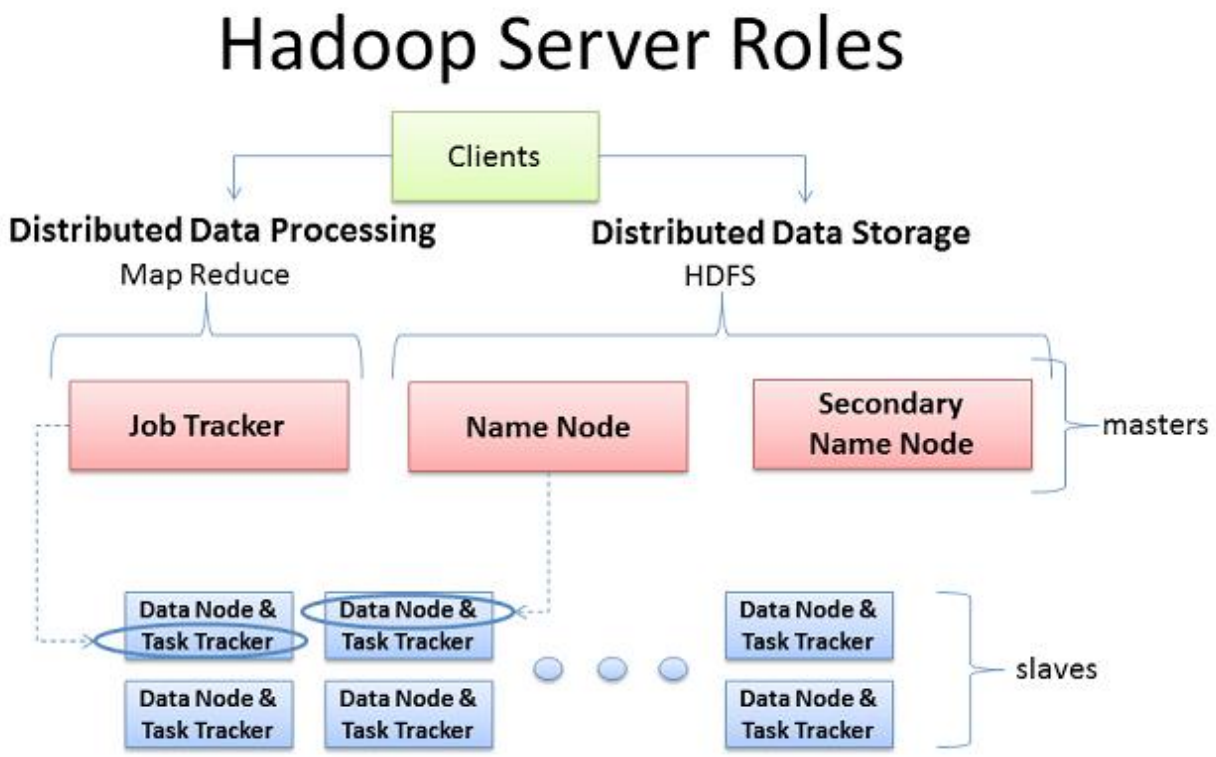
byte array layout: [counter, counter, counter, byte 1, byte 2, byte 3, byte 4, byte 5, byte 6, byte 7, byte 8, byte 9, byte 10, byte 11, byte 12, byte 13, byte 14, byte 15, byte 16, byte 17]

Google Glass- Progress

Previous progress: The android Bluetooth APIs have been implemented on the Glass and the phone and the data transmission of very large files is about done. Separately, the MobilityFirst client code has been written for C++ and needs to be implemented in Java for android.

Future Goals: The camera data collection needs to be implemented on the Glass and the MobilityFirst client code needs to be converted to be used for android. Also, a system to tell the user the results of the image recognition algorithm needs to be designed.

Hadoop Framework



http://bradhedlund.com/2011/09/10/understanding-hadoop-clusters-and-the-network/#download

Hadoop Framework

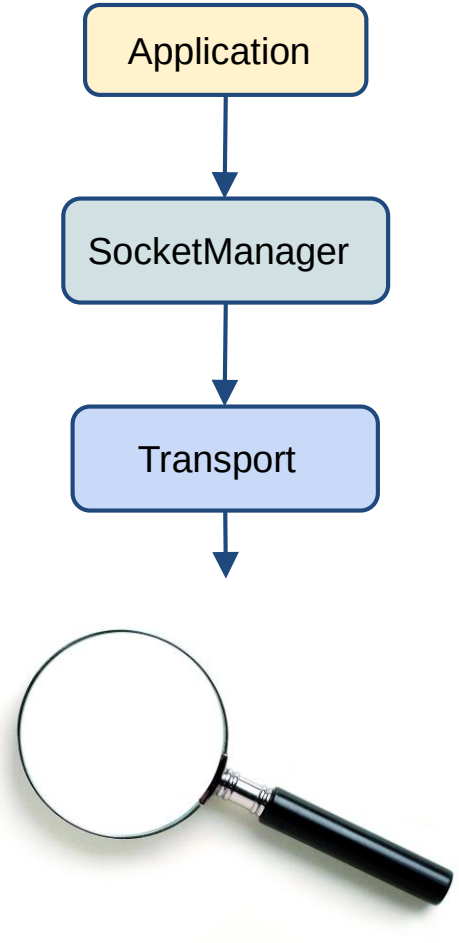
Background: present database size: 100 images vs practical requirements: over 5,000 images

Current Progress: Configure Hadoop Environment on sb5 including two nodes,
one for master, one for slave.

Challenges: Incompatible coding language between Hadoop framework & CPS-project
-> Implement Hadoop Streaming API

Data Encryption

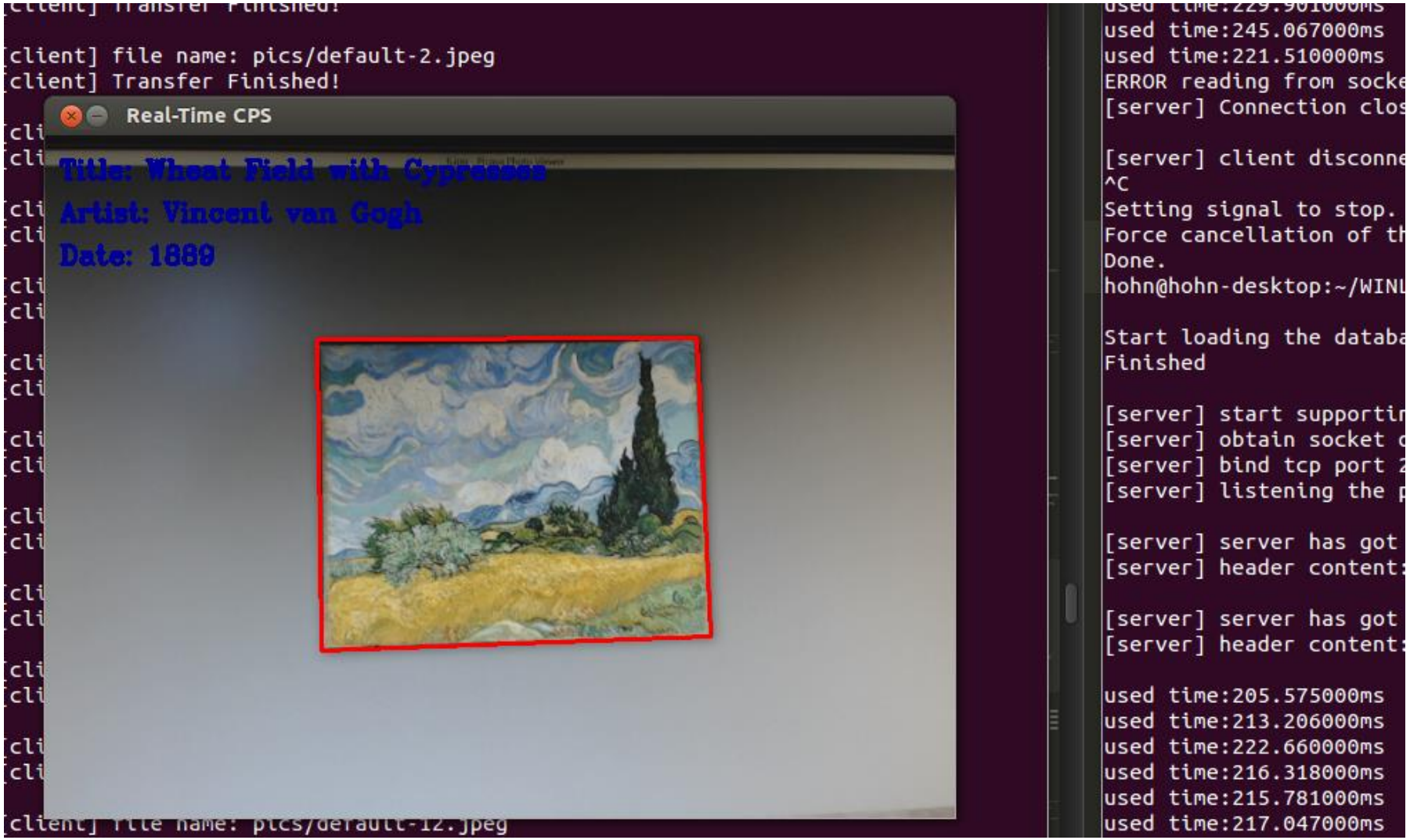
- Data at transport layer is split into blocks
- GUID serves as public key
- Cipher used depends on block size



OpenSSL

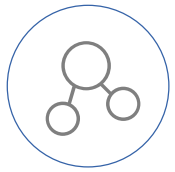
- An open-source implementation of SSL/TLS
- Secure transmission of large amounts of data
- Libraries provide cryptographic/utility functions
- Capable of hosting multiple ciphers

DEMO for Museum Scenario



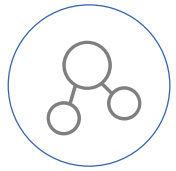
This is the TCP version. The client side is now displaying the result, including title, artist and date of the paintings.

Next Week Plan



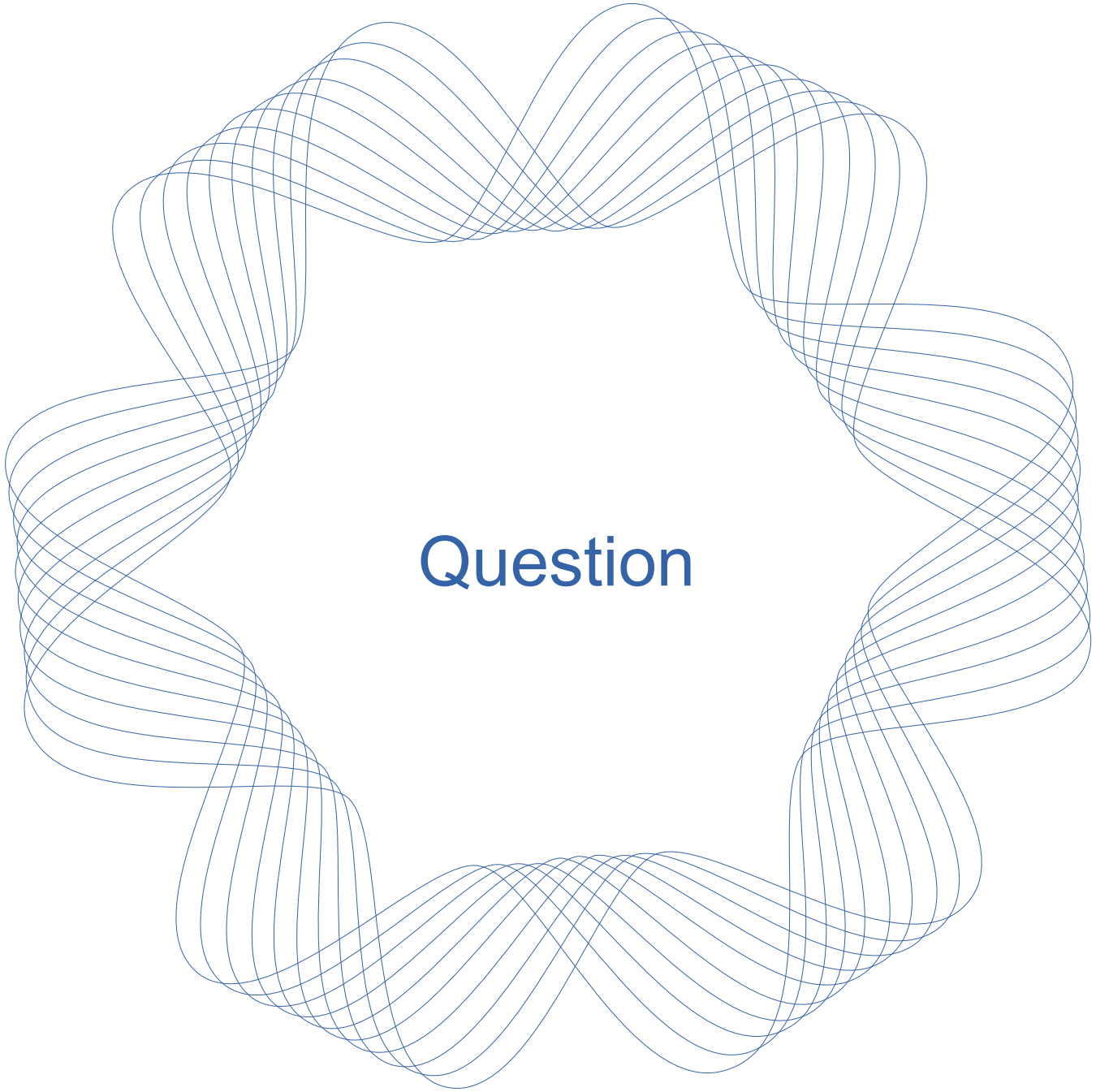
Set up the connection between mf network and the general computer through an access point.

Develop the client program based on google glass and Android phone.



Improve the speed and accuracy of the recognition on Image processing aspect and on the cloud computing aspect.

Try to implement optimal data encryption using cryptographic/utility functions from the OpenSSL core library.


Question