Suman Maroju
https://github.com/Maroju100/Resume/blob/master/Projects.pdf

PROJECT DETAILS: DEVELOPMENT WORK DONE AT VARIOUS ORGANISATIONS (2010-2018)

INDIGIS3D A GRAPHICS GIS (GEOGRAPHIC INFORMATION SYSTEM) at Vizexperts India Ltd (2010-2011)

A GIS software developed by us (a team of 5) at Vizexperts, India, Pvt. Ltd for CAIR, DRDO. The figure 1 below shows different filters implemented for performing various scientific computations on raster and vector files like Dem data, population or any other statistical data over terrain.

Languages and tools used: C++, Openscenegraph, Opengl, shell scripting, buildbot, CVS

Figure 1: Snapshot of INDIGIS3D software developed.

VENNFER HD: H.264 and G.729 BASED CODEC DEVELOPED FOR REAL TIME VIDEO CONFERENCING FOR INTELLISYS TECHNOLOGIES & RESEARCH LTD: (2013-2014)

Figure 2. Snapshot of "VennferHD" Video conferencing software.

Worked in a team of 4 for the development of a video conferencing software based on H.264. Figure 2 shows a snapshot of Video conference system developed. I did various performance improvements in Video and Audio streaming. Worked on RTP and multicast protocols for streaming over network in unicast and multicast scenarios. Did various QOS optimizations for the improved streaming performance.

Languages and tools Used: C, C++, VP8, Directshow, RTP, Multicast, G.729, H264 format

RBHMS: RAILWAY BRIDGE HEALTH MONITORING SYSTEM BASED ON WIRELESS SENSOR NETWORKS FOR RDSO, INDIAN RAILWAYS (2014-2016).

Developed an energy efficient system using an SBC (Single board Computer- RaspberryPi) to automate event based data collection and transfer of sensor data from Indian Railway Bridges using IoT protocols in constrained networks. The user interface and sensor management is entirely developed by me. (Figure 3-4). Also worked extensively on the Machine learning aspects and signal analysis for damage detection in bridges from the accelerometer and strain data obtained from the sensors attached to the bridges.

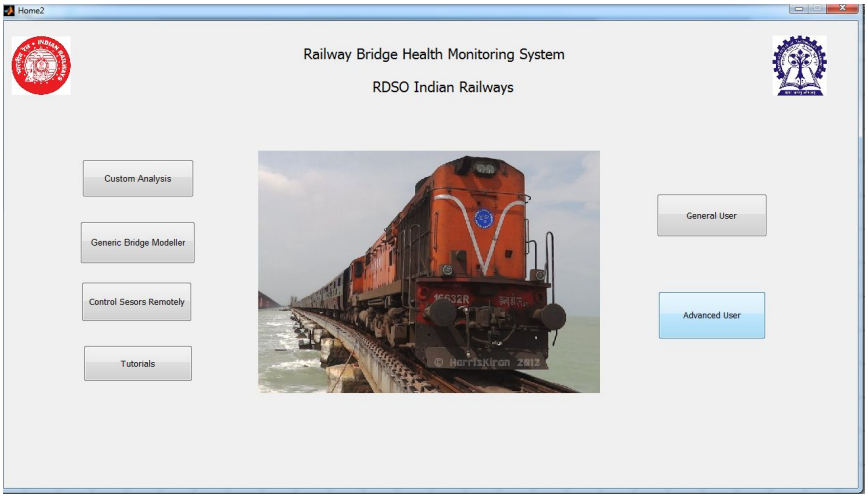

Figure 3. User Interface RBHM system developed for Indian Railways.

The **RBHMS** software allows various signal processing and machine learning algorithm to be applied to sensor data for detection of damages and analysis of current state of the structure. The software also assists in modeling the structure efficiently.

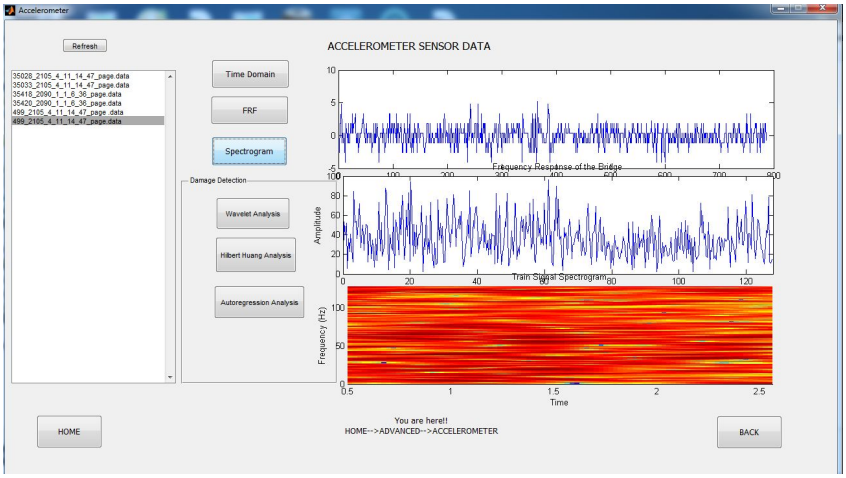

Figure 4. Sensor Analysis Tab for RBHM system.

Automated event triggered wireless sensor data collection system developed for the **RBHM** project is shown in left figure 5. Figures in Right shows Screenshot of CoAP protocol based Client and Server used for sensor data transfer from constrained networks to remote analysis server.

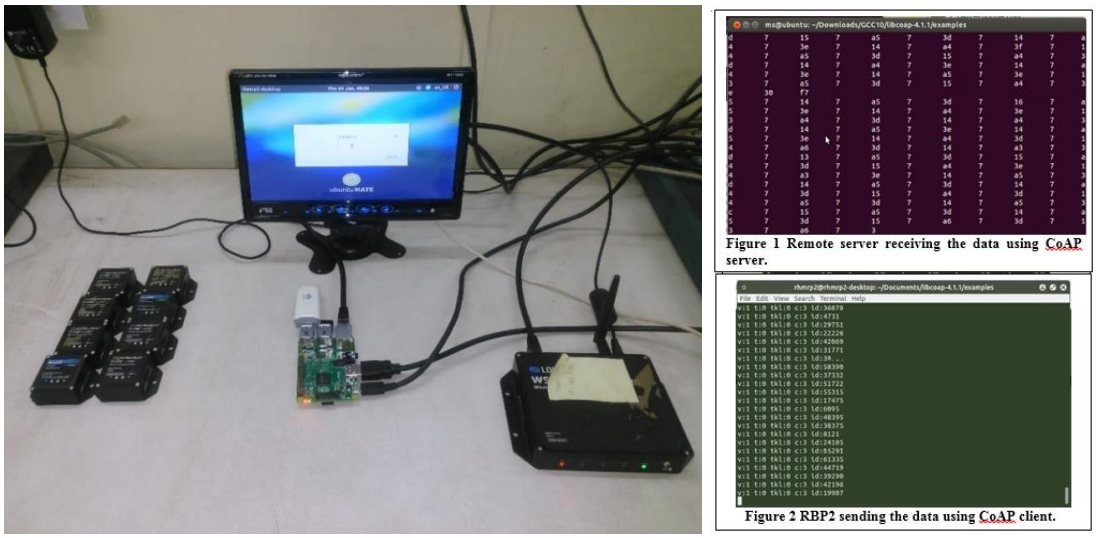

Figure 5. RBHM system developed for the project.

Languages and tools used: ASM, C, Matlab, RasberryPi, Accelerometer and Strain sensors (WSDA 1500, G link sensors), Coap protocol, Autoregression analysis, moving averages, GSM.

3 TIER FLOOD WATER MONITORING SYSTEM SOFTWARE DEVELOPED FOR INDIAN RAILWAYS:

An SMS based alerting system developed for Indian Railways to monitor flood levels and send 3 tier SMS based on the water level detected by radar based water level sensor.

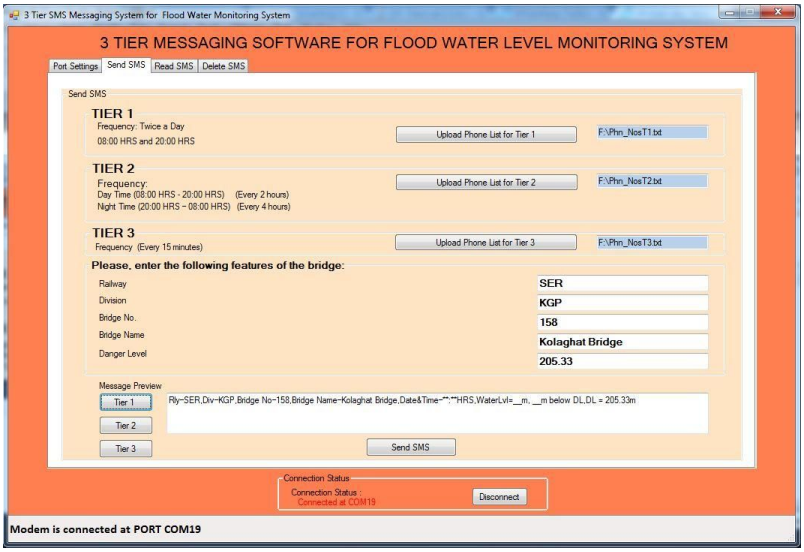

Figure 6. Automatic 3 Tier Messaging System for Flood Monitoring.

Languages and tools Used: C# and SMS AT commands

PROJECTS DONE AT NORTHWESTERN UNIVERSITY (2016-2018)

SDN BIG DATA FAULT ANALYSIS- DIAGNOSIS FRAMEWORK & DATA COLLECTION (2016 SEP-2016 DEC)

Project done in a team of two on SDN big data fault analysis. Project involved setting up of the opendaylight boron and associated frameworks for fault analysis such as TSDR, Hadoop, Spark HBASE, Cassandra for the collection and analysis of various networking faults and statistics. Various traffic statistics collected include OpenFlow Statistics, Flow Stats, Flow Table Stats, Meter Stats, Group Stats, Port Stats, Queue Stats, SNMP, NetFlow, sFlow, Syslog, Controller Metrics

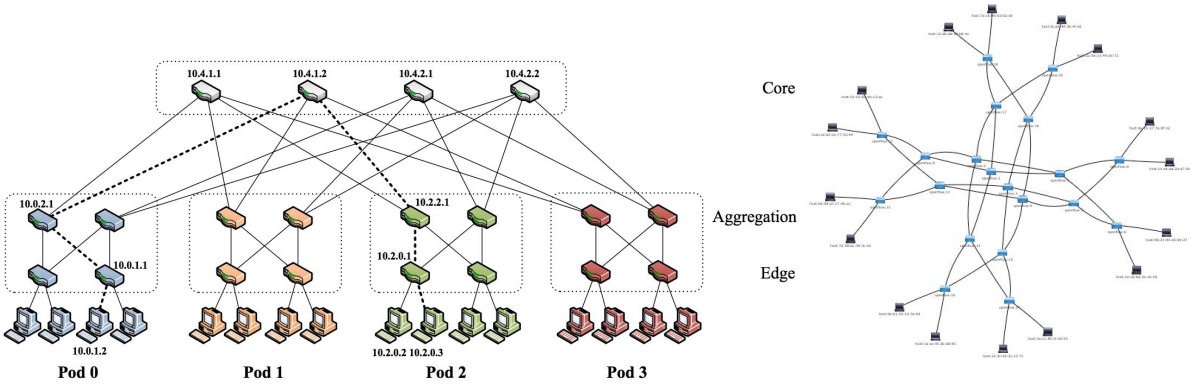

Figure 9. Mininet based network with 4 core switches, 8 aggregate and 8 edge switches and 16 hosts used for the project.

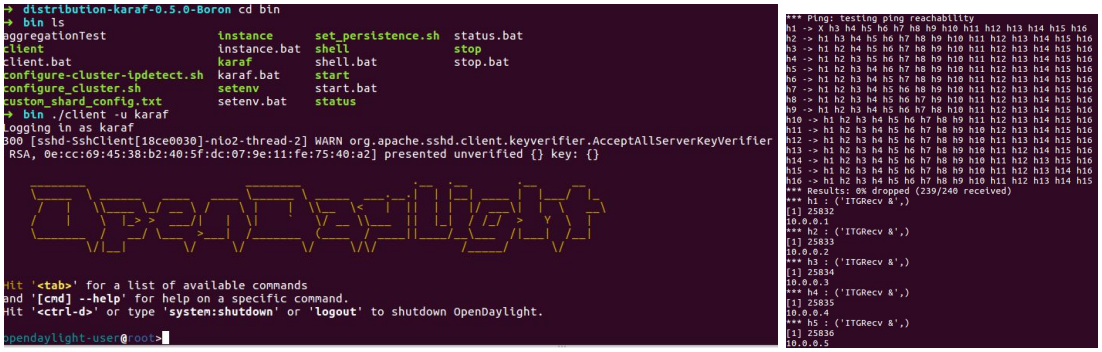

Figure 10. Opendaylight based SDN collecting various network statistics (Layer-4 data collected.protocols: TCP, UDP, ICMP, DCCP, SCTP, Layer-7 data collected, Predefined stochastic PS (Packet Size) and IDT (Inter Departure Time) profiles:CounterStrike (active and inactive), VoIP G.729)

Languages and tools used: NFV, SDN, mininet, Hadoop, Spark, Hbase, Cassandra and traffic generators

BIKE WATCHDOG: AN ANDROID BASED SYSTEM FOR DETECTION OF BICYCLE THEFT AND LOCATING BASED ON LIVE LOCATION UPDATES (2017 APRIL- 2017 JUNE)

Figure 7 shows the interface developed on android device to input the user phone number and the bluetooth MAC address on the device attached to the bike. The figure shows the sms received at the phone when the bike moves (indicating possible theft).

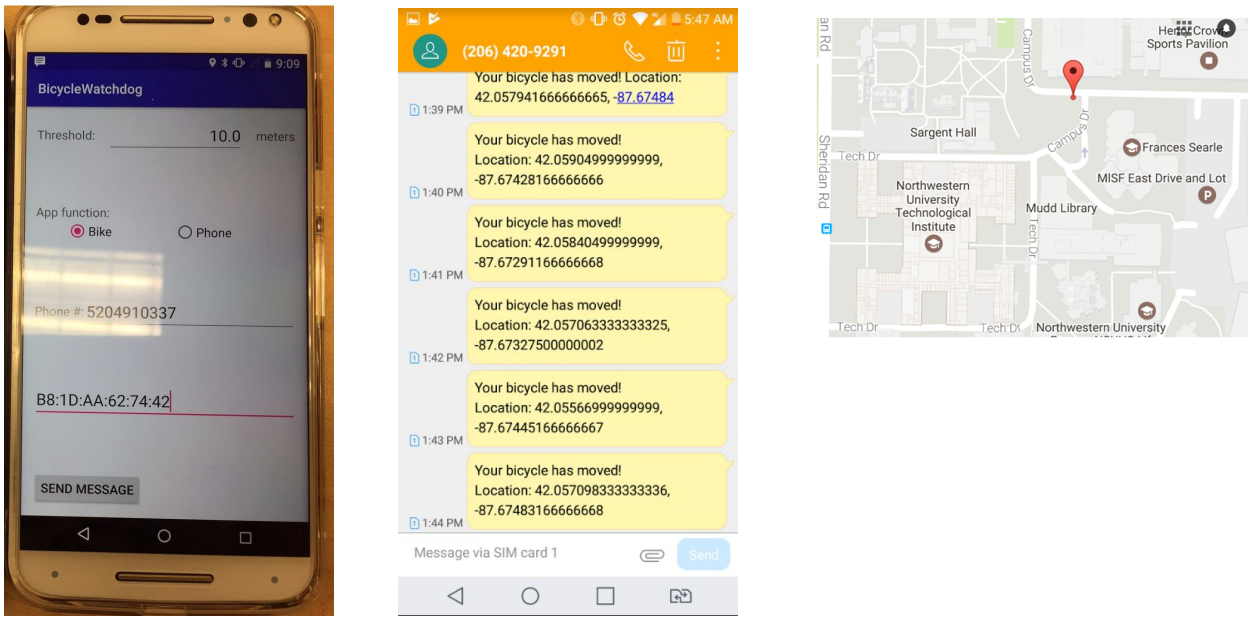

Figure 9. First figure from left shows the view at the device attached to the bike and the second figure shows the live sms alerts received at the user phone. Third figure shows the location of the moved bike plotted on the google maps.

Languages and tools used: Android, Google location API, Bluetooth manager, SMS API.

https://github.com/aiqiliu/Bike-Watchdog

SAFEMAP: A CRISIS MANAGEMENT APP DEVELOPED FOR DESKTOP AND MOBILE DEVICES

The project allows the users in crisis such as floods, earthquake etc to report their location and and visualize crisis using the heatmaps on a web or mobile device. Built using Javascript, HTML and CSS.

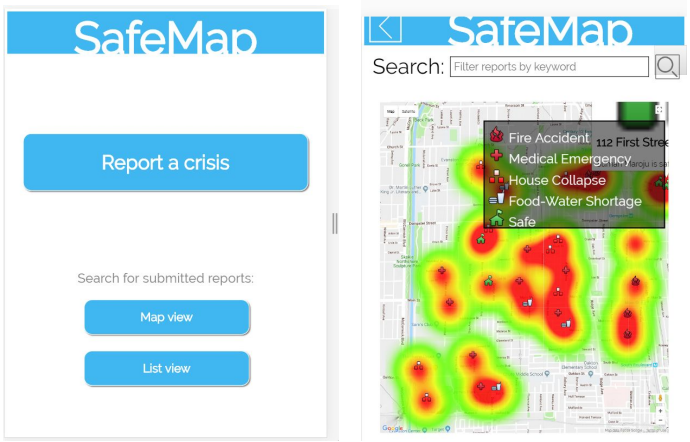

Figure 8. Main page and Map view page of the SafeMap app.

Languages and tools used: Javascript, CSS, HTML, Google location services.

Github link: https://github.com/brotatotes/SafeMap

TEAMUP PROJECT: OPTIMAL TEAM ALLOCATION SYSTEM BASED ON IBM WATSON PERSONALITY INSIGHT API (2017 Sep-2017 Dec)

Worked in a team of 3 for the development of a AI based optimal team allocation system that can group a set of individual into teams such that each team is balanced in term of more than 70 personality characteristics obtained from IBM Watson personality insight API. Wrote round robin based team allocation algorithm in python and worked in the implementation of web interface using Flask.

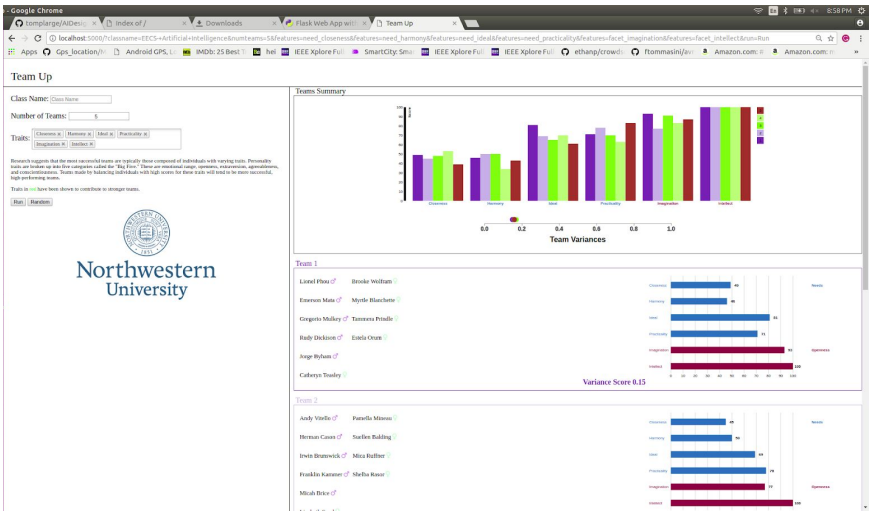

Figure 7. TeamUp web interface to upload team information and create optimal teams and view each team statistics.

Languages and tools Used: Python, Flask and Web technologies (HTML, CSS, Javascript)

IMAGE CLASSIFICATION: BUILD AN IMAGE CLASSIFIER BASED ON CONVOLUTION NEURAL NETWORKS (USING TENSORFLOW)

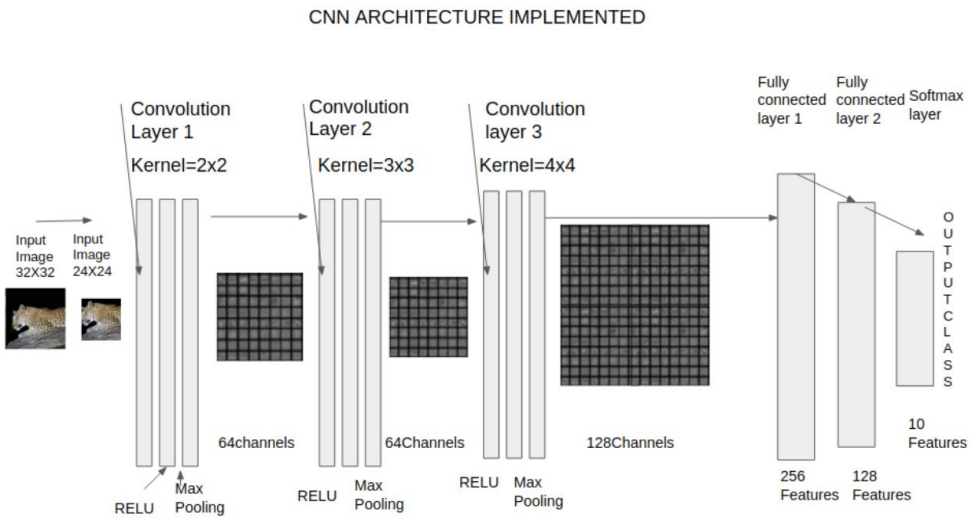

Figure 8. CNN Architecture for image classification.

Building and training a CNN model using TensorFlow API for image classification based on deep learning principles and optimization techniques for improving the image classification accuracy. Used a pre trained Google inception v3 model for classifying images and compared the classification results with the model built.

Github link: https://github.com/Maroju100/CNN-image-classificatoin.git

TRIGCAM: A SMARTWATCH TRIGGED WEARABLE CAMERA SYSTEM FOR EFFICIENT FEEDING HABIT MONITORING (2017 June-2018 March)

Developed a system called TrigCam for feeding gesture monitoring of users. TrigCam system consists of a smartwatch paired with an android smartphone to trigger the recording of the videos at a raspberry pi based fisheye camera (TrigCam). TrigCam records the feeding videos from a first person perspective only when feeding gestures are detected at the smartwatch worn by a user. Smartwatch triggers the phone and phone relays the triggers to the TrigCam to start 20 seconds of video recording at TrigCam for a trigger. Multiple consecutive triggers occuring in 20 seconds window are considered as a single Trigger to avoid backlogging of triggers.

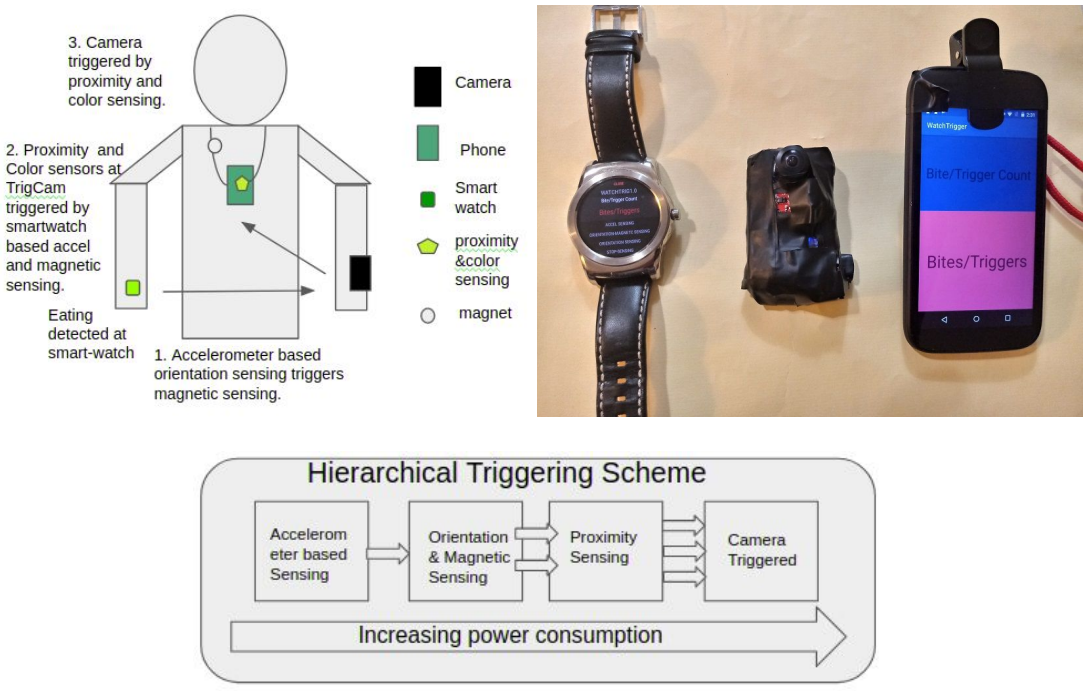

Figure 9. Smart Watch Triggered Wearable Camera System

Languages and tools used: Android, python, Java, RasberryPI zero, Orientation, Magnetic and Accelerometer sensors of a smartmatch, sensor fusion based on google android API,, bluetooth, Wifi scripts, Syncthing, Fish eye lens, OPENCV, Tensorflow and weka for training the system.

Github link: https://github.com/Maroju100/TrigCam