SE/EE/CPRE 491
Project Title: Economic Home Security System
Group No: 42
SD-MAY20- Group 42

Weekly Report 2

TEAM MEMBERS:
Uma Abu
Andrew Tran
Sohum Sawant
Kamini Saldanha
Merin Mundt
Lucas Jedlicka - Project Lead

Weekly Summary: We met with our advisor/clients and he gave us some tasks to accomplish.

Past Week Accomplishment: The tasks given to us were to define scalability for our application. This involves discussing the tradeoffs between cost, efficiency and multiple devices.

The next task we were given was to create scenarios for our application. We were tasked to create the most simple end to end scenario for our home security system to run through. Through this process, we can identify some problems we didn't know had and come up with more use cases.

We were also asked to create the justification for the term "Economic" in the title of our project "Economic Home Security System".

Issue: Justification for the term "Economic" in the title "Economic Home Security System"

Three main factors contribute to the economic viability of this system.

- Optimized bandwidth usage
- Minimal GPU usage server side
- Minimized storage

Bandwidth is reduced for clients viewing streams by transcoding the high-resolution stream from the camera, down to a configurable (default low) resolution. Bandwidth is reduced in the phone camera device case via minimized frame rate and only streaming after the local device detects motion. The motion detected stream will be rejected once the object detector deems the contents undesirable (ex anything other than a human or car in the region of interest).

GPU usage on the processing server is a couple ways. Usage may be reduced by only transcoding streams when a client device needs them. Another is by a priority queue pooling system of multiple streams, only doing object detection on individual frames every X seconds. Real time object detection is not required for push notifications; the first frame from every new stream will be placed at a high priority.

Storage is to be reduced by being conservative with which footage is worthy of being saved. To determine whether the stream should continue to be saved a frame will be object detected at some TBD interval after the first. Once objects of interest are out of the ROI, a counter begins counting down until another object of interest comes into frame (resetting the counter) or it hits zero. At zero the clip will end. Contingent on the formats we receive (MJPEG is huge), the completed clip will be converted to H264 and stored. Upon viewing the stored video, the user will be strongly encouraged to delete the clip via UX cues, and after some amount of weeks old clips will be purged. If a user desires to keep a clip, they should download it.

Pending Issues: Scaling, determining required hardware, and researching tradeoffs that will make our system “Economic”.

Individual Contributions:

Name	Contribution	Hrs this week	Total hrs
Uma Abu	Researched motion detection on the front end part with some javascript library.	3	6
Lucas Jedlicka	Played with tensorflow and yolonet object detection. Created scenarios to justify design choices.	4	7
Sohum Sawant	Started learning Python to be able to develop the backend. Started working to be able to receive streams via sockets.	4	7
Merin Mundt	Started talking about the client application and what should be required. Looked at what requirements are more important than others.	3	3

Kamini Saldanha	**Researching what to prioritize in terms of scalability for the project in terms of cost, efficiency and multiple devices. Started looking into features and price ranges for different graphic cards for server. Looked into capabilities for motion detection library in JavaScript to reach scalability decisions.**	**3**	**6**
Andrew Tran	**Worked on creating cases for a simple end-to-end scenario and go through defining the scope. Have not further investigated backend work.**	**3**	**6**

Plans for the upcoming week: We plan to start developing a prototype for our simple end to end scenario. We need to create GitLab issues and assign tasks.