Title: Visual Diary

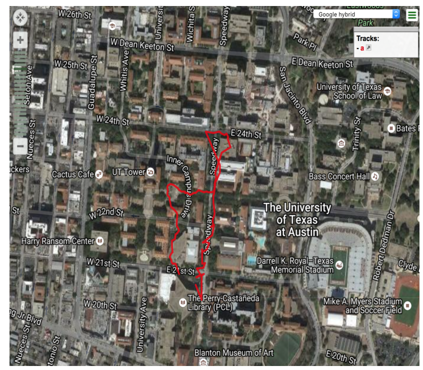
Group Members: Kunal Lad, Rahul Huilgol, Rohan Tripathi, Saharsh S Oza

Accomplishments:

1. **Developed android Application to collect data:**
   The project requires location, acceleration and video to generate a daily summary. Mobile phones provide GPS, accelerometer and camera with microphones readily available. Unfortunately we found no existing apps to gather this data. So we created a custom Android App for recording the required data. The app allows you to capture data using the phone's accelerometer, GPS and camera. We decided to take video instead of still images which gave us audio data as well.

2. **Data Collection:**

   We collected data in the campus. Each dataset example corresponds to a 20 minute journey of a person carrying their mobile phone. The video is collected at 24 frames per second. GPS and accelerometer data is logged only when the metric value changes. This is important for storage but makes subsequent analysis tricky.
   Metrics Collected:
   a) Acceleration across x,y and z coordinates
   b) Latitude and longitude in GPS
   c) Video at 24 fps
   Sensors used: Accelerometer, GPS, Phone's Camera

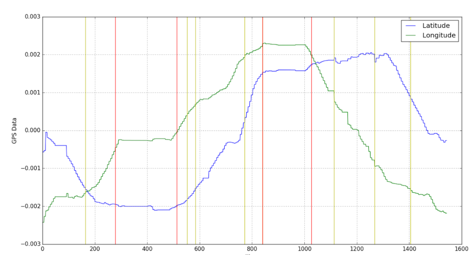


The above image represents an example of the path we took to record data through the app.

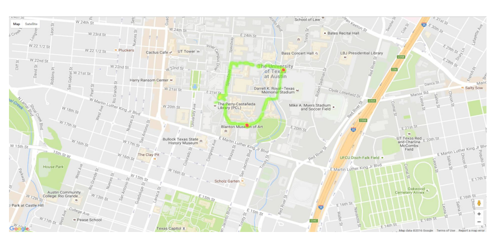
3. **Identifying stationary locations:**

   The first task in being able to generate a diary is being able to identify:
   i) GPS coordinates where the user is stationary.
   ii) the time at which the user is stationary at that location.
   In order to achieve this, we followed a 2 level approach. We used a **slope (rate of change)** of less than 0.0001 (empirically arrived at) of the coordinates to identify regions of interest. Next we used **standard deviation** with **variable frame length** in that region to find narrow down the identification.

   Note: Both of these metrics are evaluated on frames with minimum size of 150 seconds. 150 seconds is the threshold above which can a coordinate qualifies to be stationary.
   To test the accuracy of our approach, we generated heat maps of the locations identified as stationary using gmap library in python. The 2 dots in the map accurately identify the areas where the user had been stationary.



Plot of normalized latitude and longitude across time. The yellow line demarcates the region of interest using a slope threshold and the red line demarcates the final selection using standard deviation.



The figure shows heat maps corresponding to the 2 red regions marked in the graph above.

Challenges:

1. **Android Data Collection:**
   Collecting data from all 3 sensors using the android SDK posed the following challenges:
   i) **Identifying sampling rate** of each of the sensors. Too quick would mean a lot of redundant data and too slow could result in information loss. After a few iterations we settled at 5 second samples.
   ii) **Memory vs Resolution in camera data:** Very high resolution and fps would have resulted in a GB of data within 20 minutes. So we had to identify which combination of fps and resolution would result in storage of less than 300 MB. Our final config was 24 fps with 720p that resulted in 254 MB of video in 20 minutes.
2. **Narrative camera has been discontinued:** We started the project with the idea of using the Narrative camera for our project but, unfortunately, the company discontinued support for it. This meant we had to design the data collection app by ourselves.
3. **Identify stationary location using GPS data:** This was a challenging problem to solve for many reasons.
   i) Data had to be normalized to even visualize any perceivable variation.
   ii) Identify thresholds for slope and standard deviation to classify stationary vs non stationary.
   iii) Slope had to be used to identify regions with minimal movement. But this would allow slow moving users to be wrongly classified as stationary. Hence standard deviation was used inside the regions demarcated with slope metric. The deviation was computed on an adaptive frame size starting from 150 to the limit of the slope demarcation bucket.
4. **Inability to obtain SenseCam32 dataset**: This dataset was annotated with first person description of daily activities. The authors had initially agreed to share the dataset but have not responded since. Hence the RNN used to generate text for the images will be 3rd person in place of first person descriptions.

Updated Timeline:

Week	Task
1st November	1. Generate text to describe GPS classification. 2. Classify accelerometer in buckets of fast and slow movement 3. Use this to segment the video differently for each of the segments
8th November	1. Be able to identify salient images from a sampled set of images
15th November	1. Use neural Talk to generate third person text for images selected above
22nd November	1. Dashboard to represent: a. GPS trail on a map b. Images in each leg of journey/stationary location c. Text to summarize movement and images

Updated Deliverables:

Since the SenseCam 32 dataset is unavailable, there will be no first person account of data to train the RNN. So pre trained Neural Talk will be used. This will mean the textual representation of the image will not be a first person account. Eg: ‘I am working’ will instead be ‘A laptop on a table.’