Data Collection Manual

Yanning Li, Fangyu Wu, Will Barbour
Dec 4th, 2015

This manual describes the data collection procedure, data post-processing, and documentation. All future data collection should follow the specified data format to avoid rewriting the file parser. All collected data must be well organized and documented for easy access.

1. Data Collection Procedure

All future data collection should be conducted following the procedure:

1. One-page plan including:
   - Where, when, who, and how long?
   - Objective. Why is this data collection necessary?
   - What data will be collected, e.g. PIR, Ultra, IMU.
   - The preparation procedure which includes testing the sensors, communication, and processing in the office. How long will this take.
   - The proposed data collection setup in the field.
2. Data collection in the field.
3. Post-processing:
   - Convert the raw PIR data from HEX to temperatures in Celsius.
   - Combine all data into one csv file: PIR, Ultrasonic, IMU, Ground truth label data.
4. Data organization and documentation. Each data set should be organized in the following structure:

```
./dataset_20151204_152334
----./plan.pdf
----./readme.pdf
----./20151204_152334.csv
----./code/
----./raw_data/
     ----./raw_data_files.csv
     ----./post_process.py
     ----./video_20151204_152334.mp4
```

The folders and files are explained as follows:

`./dataset_20151204_152334`

This is the main folder containing the data set. Name it by the first timestamp of the data. The date time format in python is `%Y%m%d_%H%M%S`

`./plan.pdf`

The one-page plan written before the data collection

`./readme.pdf`

A one-page similar to the `plan.pdf` with the following additional information:

- Any changes from the plan made in the field data collection due to hardware failures or environmental causes.
- The data format of the aggregated data file `20151204_152334.csv`

`./20151204_152334.csv`

This is the aggregated file for all the data including the PIR, IMU, USON, and LABEL in the format specified in the next section. If there are multiple separated data collection experiments, save the aggregated data for each experiment in a separated file.

`./code/`

This folder contains the firmware and other code used in the field data collection. E.g. the Arduino firmware, the raspberry pi script, and the laptop script.

`./raw_data/`

This folder contains the raw data collected.

`./raw_data_files.csv`

These are the raw data files received from the Serial and saved in the laptop.

`./post_process.py`

The python script for post processing of the raw data, including converting the raw HEX to temperature in Celsius and aggregating all data files into one data file.

`./video_20151204_152334.mp4`

This is the video during the data collection.

2. Data Format

All data should be aggregated and saved in a single file `20151204_152334.csv`

- Each row is a slice of data of different types. Items in the row are separated by comma with no space. The format of each row is as follows:

Type_of_data,timestamp,data

type_of_data, constant strings:

PIR_1x16_2nd: the 1x16 pixel of the 2nd row of the center PIR sensor
PIR_4x48: all pixels for the three PIR sensors
USON: ultrasonic sensor data
IMU: the IMU sensor data

timestamp: `%Y%m%d_%H:%M:%S_%f`

data:

PIR: float array, ***vec*** of the pixel matrix in Celsius
USON: float value, in meters
IMU: float array,
(mag_x, mag_y, mag_z, accel_x, accel_y, accel_z, gyro_x, gyro_y gyro_z)
LABEL: count (1 or 2), speed (MPH)

- If a row contains corrupted data, then save as:

Type_of_data,invalid_read

Examples:

PIR_1x16_2nd,20151204_15:23:34_123456,Ta,pixel_1x1,pixel_1x2, …,pixel_1x16
PIR_2x16,20151204_15:23:34_123456,Ta,pixel_1x1,pixel_2x1,pixel_1x2, …,pixel_2x16
PIR_4x48,20151204_15:23:34_123456,Ta,pixel_1x1,pixel_2x1,pixel_3x1, …,pixel_4x16

USON,20151204_15:23:34_123456,7.3234

IMU,20151204_15:23:34_123456, 1,1,1,2,2,2,3,3,3

LABEL,20151204_15:23:34_123456,1,13.4
LABEL,20151204_15:23:34_123456,2,14