Dataset documentation for the paper "Peeves: Physical Event Verification in Smart Homes"

Abstract

This report contains the documentation of the smart home event and sensor dataset collected and used in the paper "Peeves: Physical Event Verification in Smart Homes" [1]. The dataset contains events from 12 different event sources and data from 48 sensors collected in an office environment over the course of twelve days. The report first presents the general structure of the dataset and then gives further details on each event and sensor type.

1 Dataset structure

1.1 Folder Hierarchy

The dataset for each Raspberry Pi device contains a single folder for each sensor attached to the device. Sensor data collected by a laptop can be found in the folder `laptop`, whereas the event data of the devices triggered through the tablet interface can be found in the folder of `pi5`. As an example, the folder `pi1/MPU` contains the data of the MPU6050 accelerometer/gyroscope attached to Raspberry Pi 1. The data of most sensors is split into hourly segments, with filenames in the format `YYYYMMDDhhmmss.csv`. Deviations from this format are described in the relevant sensor subsections.

1.2 File format

All files are provided in **`.csv`** format, the first row is a header that indicates the content of the columns. The column "timestamp" contains the date and time of the sensor measurement/event and all datetime values are in the format `YYYY-MM-DD hh:mm:ss.mmmmmm` and in UTC timezone. Further information can be found in the relevant subsections for each event/sensor.

1.3 Exclusions

There were a few time periods when the recorded ground truth was not correct. This happened due to network issues, calibration issues and device faults. Therefore, these time periods were excluded from the evaluation of the affected sensors. Exclusions are described in the relevant event subsections. All exclusion files have the columns `start` and `end` that determine the excluded time period. The values are in the format `YYYYMMDDhhmm`.

2 Events

2.1 Smart cam

The camera used in this experiment is a Samsung SmartCam SNH-VP6410PN with privacy mode feature. This camera can be moved into a position that prevents it from recording its surroundings.

All event data are found in a single file `laptop/Cam/cam.csv`. The camera has two event types which are determined by column `privacy`:

- **1**, when privacy mode is turned on, i.e., the camera is moved to conceal the view
- **0**, when privacy mode is turned off, i.e., the camera moves back to the previous position

2.2 Coffee machine

The coffee machine Smarter Coffee (v1), grinds beans and brews filter coffee. During this experiment, the device was controlled through a tablet interface over wifi. Users were able to use this interface to select a number of cups between 1 and 12.

All event data are found in single file `pi5/CoffeeMachine/coffee.csv`. The column `cups` is the number of cups requested through the tablet interface.

2.3 Door

The door is outfitted with a magnetic contact switch to record door opening/closing events. When the door is not propped open, it closes through an automatic closing mechanism.

The files with event data are found in folder `pi8/DOOR`. The door has two event types which are determined by column `DOOR`:

- **1**, when the door is opened
- **0**, when the door is closed

2.4 Fan

The tower fan is remote controlled through an infrared interface by a Raspberry Pi Zero which is mounted on top. During the experiment, users were able to trigger it through a tablet interface.

All event data are in a single file `pi5/Fan/fan.csv`. The column `trigger` indicates if the event was triggered manually (`manual`) or automatically as scheduled (`auto`). Start and end times of excluded time periods (due to network issues affecting ground truth) can be found in the file `exclusions/fan_exclusions.csv`. The fan has two event types which are determined by the column `state`:

- **on**, when the fan is turned on
- **off**, when the fan is turned off

2.5 Fridge

The fridge door is outfitted with a magnetic contact switch to record opening/closing events. It does not have an internal light source.

The files with event data are found in folder `pi10/FRIDGE`. The fridge has two event types which are determined by column `FRIDGE`:

- **1**, when the fridge door is opened
- **0**, when the fridge door is closed

2.6 Light

The light switch is outfitted with a force-sensitive resistor (FSR) to detect button presses. The current state is kept internally, and an event is triggered when a pre-configured threshold is exceeded.

The files with event data are found in folder `pi7/LIGHT`. Start and end times of excluded time periods (due to uncertain ground truth) can be found in the file `exclusions/light_exclusions.csv`. These were excluded because the light turned off automatically due to inactivity or because of issues with the FSR-based sensor.

The light has two event types which are determined by the column `LIGHT`:

- **1**, when the light is turned on
- **0**, when the light is turned off

2.7 PC

The PC is turned on and off automatically using wake-on-lan/SSH commands. All event data are found in a single file `pi5/PC/pc.csv`. Start and end times of excluded time periods (due to network issues affecting ground truth) can be found in the file `exclusions/pc_exclusions.csv`. The PC has two event types which are determined by the column `state`:

- **on**, when the PC is turned on
- **off**, when the PC is turned off

2.8 Radiator

The eqiva eQ3 thermostatic radiator valve is controlled over Bluetooth. During the experiment, users were able to trigger it through a tablet interface.

All event data are found in a single file `pi5/Radiator/radiator.csv`. The column `trigger` indicates if the event was triggered manually

(`manual`) or automatically as scheduled (`auto`). Start and end times of excluded time periods (due to calibration issues affecting ground truth) can be found in the file `exclusions/radiator_exclusions.csv`. The radiator has two event types which are determined by the column `thermostat`:

- **on**, when the radiator is turned on
- **off**, when the radiator is turned off

2.9 Ring

The Ring doorbell data is polled through the Ring.com Python API [1]. The files with event data are found in folder `laptop/Ring`.

2.10 Screen

The screen is turned on and off automatically using commands over SSH. All event data are found in a single file `pi5/Screen/screen.csv`. Start and end times of excluded time periods (due to network issues affecting ground truth) can be found in the file `exclusions/screen_exclusions.csv`. The screen has two event types which are determined by the column `state`:

- **on**, when the screen is turned on
- **off**, when the screen is turned off

The column `message` gives further context when the screen was turned on/off implictly by turning the PC on/off.

2.11 Window

The window is outfitted with two magnetic contact switches to record opening/closing events. It is a sliding window consisting of two separate window halves.

The event data for the left side of the window are found in folder `pi4/WIN1`. They were not used in the paper due to lack of use during the run of the experiment. The event data for the right side of the window are found in folder `pi3/WIN2`. The window has two event types which are determined by column `WIN2`:

- **1**, when the window is opened
- **0**, when the window is closed

[1] https://github.com/tchellomello/python-ring-doorbell

2.12 Window shade

The Teptron Move window shade controller is controlled over Bluetooth. During the experiment, users were able to trigger it through a tablet interface.

All event data are found in a single file `pi5/WindowShade/shade.csv`. The column `trigger` indicates if the event was triggered manually (`manual`) or automatically as scheduled (`auto`). Start and end times of excluded time periods (due to calibration issues and device faults affecting ground truth) can be found in the file `exclusions/shade_exclusions.csv`.

The window shade has two event types which are determined by column `position`:

- **up**, when the window shade is fully opened
- **down**, when the window shade is fully closed

3 Sensors

The following subsections describe the sensors in more detail. Sensors were not calibrated before the start of the experiment.

3.1 BMP280

The BMP280 air pressure/temperature sensors are attached to Raspberry Pis and their data is polled over the Raspberry Pi's I^2C interface. The sensor data are found in the folder `pi<X>/BMP`, where `<X>` is the number of the Raspberry Pi the sensor is attached to.

The csv files have the following columns:

- **temperature** (°C)
- **pressure** (hPa)

3.2 BME680

The BME680 air pressure/temperature/humidity sensors are attached to Raspberry Pis and their data is polled over the Raspberry Pi's I^2C interface. The sensor data are found in the folder `pi<X>/BME`, where `<X>` is the number of the Raspberry Pi the sensor is attached to. The only exception are sensors for the fridge. The data for the sensor inside the fridge can be found in the folder `pi10/BMEIN`, the data for the sensor on top of the fridge can be found in the folder `pi10/BMEOUT`,

The csv files have the following columns:

- **temperature** (°C)

- **pressure** (hPa)
- **humidity** (RH%)

3.3 MLX90640

The MXL90640 thermal cameras are attached to Rasberry Pis and their data is polled over the Raspberry Pi's I^2C interface. The sensor data are found in the folder `pi<X>/MLX`, where `<X>` is the number of the Raspberry Pi the sensor is attached to.

These thermal cameras have a resolution of 32x24 pixels, but only the column and row averages of 24x24 pixels were recorded due to privacy considerations. All column and row average values are in °C.

3.4 MPU6050

The MPU6050 accelerometer/gyroscope sensors are attached to Rasberry Pis and their data is polled over the Raspberry Pi's I^2C interface. The sensor data are found in the folder `pi<X>/MPU`, where `<X>` is the number of the Raspberry Pi the sensor is attached to.

The csv files have the following columns:

- **accelerometer_{x,y,z}** (m/s^2) for each axis
- **gyroscope_{x,y,z}** ($°/s$) for each axis
- **temperature** (°C)

3.5 Power plugs

The TP-Link HS110 power plugs have an integrated energy monitor sampling at 1Hz. During the experiment, they were continuously polled by a laptop. Each power plug powered one of the corded devices. Some of the plugs used an older firmware version. Therefore, there are some differences regarding the data units as described below.

3.5.1 Camera plug

The sensor data for the plug attached to the smart cam are found in the folder `laptop/CamPower`.

The csv files have the following columns:

- **current** (mA)
- **voltage** (mV)
- **power** (mW)
- **total** (Wh)

3.5.2 Coffee plug

The sensor data for the plug attached to the coffee machine are found in the folder `laptop/CoffeePower`.

The csv files have the following columns:

- **current** (A)
- **voltage** (V)
- **power** (W)
- **total** (kWh)

3.5.3 Fan plug

The sensor data for the plug attached to the tower fan are found in the folder `laptop/FanPower`.

The csv files have the following columns:

- **current** (mA)
- **voltage** (mV)
- **power** (mW)
- **total** (Wh)

3.5.4 Fridge plug

The sensor data for the plug attached to the fridge are found in the folder `laptop/FridgePower`.

The csv files have the following columns:

- **current** (A)
- **voltage** (V)
- **power** (W)
- **total** (kWh)

3.5.5 PC plug

The sensor data for the plug attached to the PC are found in the folder `laptop/PCPower`.

The csv files have the following columns:

- **current** (mA)
- **voltage** (mV)
- **power** (mW)
- **total** (Wh)

3.5.6 Screen plug

The sensor data for the plug attached to the Screen are found in the folder `laptop/ScreenPower`.

The csv files have the following columns:

- **current** (mA)
- **voltage** (mV)
- **power** (mW)
- **total** (Wh)

3.5.7 Shade plug

The sensor data for the plug attached to the window shade controller are found in the folder `laptop/ShadePower`.

The csv files have the following columns:

- **current** (mA)
- **voltage** (mV)
- **power** (mW)
- **total** (Wh)

3.6 Received signal strength (RSS)

The basis for the RSS measurements are periodic access point beacons that are being captured using Ralink RT5370 WiFi dongles and tshark on Raspberry Pis. The data are found in the folder `pi<X>/RSS`, where `<X>` is the number of the Raspberry Pi the data was captured on.

The csv files have the following column:

- **signal_dbm**, received signal strength (in dBm)

3.7 SGP30

The SGP30 air quality sensors are attached to Raspberry Pis and their data is polled over the Raspberry Pi's I^2C interface. The sensor data are found in the folder `pi<X>/SGP`, where `<X>` is the number of the Raspberry Pi the sensor is attached to.

The csv files have the following columns:

- **co2**, equivalent calculated CO_2 concentration (in ppm)
- **voc**, volatile organic component concentration (in ppm)

3.8 SenseHAT

The SenseHAT sensor board is attached to a Raspberry Pi and its data is polled over the Raspberry Pi's I^2C interface. The sensor data are found in the folder `pi8/SH`,

The csv files have the following columns:

- **accelerometer_{x,y,z}** (G) for each axis
- **gyroscope_{x,y,z}** (rad/s) for each axis
- **magnetometer_{x,y,z}** (μT) for each axis
- **yaw, pitch, roll**, derived values
- **pressure** (hPa)
- **temperatureP**, temperature of air pressure sensor (in °C)
- **humidity** (RH%)
- **temperatureH**, temperature of humidity sensor (in °C)

3.9 Sound pressure levels

The sound pressure levels were measured using USB microphones. The values are the root-mean square of the audio signal fragment taken with a sampling rate of 120Hz. The sensor data are found in the folder `pi<X>/SPL`, where `<X>` is the number of the Raspberry Pi the microphone is attached to.

The csv files have the following columns:

- **rms**, root-mean square of the audio signal
- **rms_db**, root-mean square of the audio signal (in dB)

3.10 TSL2561

The TSL2561 light sensors are attached to Raspberry Pis and their data is polled over the Raspberry Pi's I^2C interface. The sensor data are found in the folder `pi<X>/TSL`, where `<X>` is the number of the Raspberry Pi the sensor is attached to.

The csv files have the following columns:

- **lux**, visible light spectrum (in lux)
- **full**, full spectrum (in lux)
- **ir**, infrared spectrum (in lux)

4 Citing this work

The dataset has been collected and first used for the paper in [1]. If you use this dataset we ask you to cite the paper:

```
@inproceedings{sbirnbach2019,
title={Peeves: Physical Event Verification in Smart Homes},
author={Birnbach, Simon and Eberz, Simon and Martinovic, Ivan},
booktitle={Proceedings of the 2019 ACM SIGSAC Conference on
Computer and Communications Security},
year={2019},
organization={ACM}
}
```

References

[1] Simon Birnbach, Simon Eberz, and Ivan Martinovic. Peeves: Physical event verification in smart homes. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2019.