Individual Report for COMP3516

1 TASK P-1

What I have done (including how to acquire ACF and the shape):

(1) Set the CSI sampling rate = sampling rate / frame length.
(2) Determine the number of window and samples per window by using the window size and the overlap size.
(3) Initialize the time vector, lag vector and acf matrix.
(4) Iterate through the subcarriers. For each carrier, iterate through the CSI matrix by window size and overlap size.
(5) For each window, calculate the ACF, normalized it and store it in the acf matrix.

2 TASK P-2

What I have done (including how to acquire MS and the shape):

(1) Calculate motion statistics by retruning first array of the lag of the ACF matrix.
(2) Calculate the ms_bar by averaging the motion statistics across all subcarriers.

3 TASK P-3

Method for combing ACF (including reasons):

(1) Use column mean of MS as weights, so that the subcarriers with higher ACF mean have higher weights
(2) Select top 10 subcarriers since less than 10 subcarriers have proper bessel curve shape based visualization of the data available, ie. topK = 10
(3) Return the indices of the top 10 subcarriers, as topK_carrier.
(4) Smooth the ACF data to remove noise by using savgol filter.
(5) Initialize ACF matrix.
(6) Iterate through the topK_carrier. For each carrier, find the sum of the smoothed acf to the combined_acf; and finally the sum of the MS to the combined_ms.

Peak Detection Design (including reasons):

(1) Use find_peaks function from scipy to detect the peaks in the combined_acf. The parameters have to be taken great care of:
    (a) height: the minimum height of the peak. It is set to be range between 0.01 and 0.5 of the max(combined_acf).
    (b) distance: the minimum distance between the peaks. It is set to be 10.
    (c) prominence: the minimum prominence of the peak. It is set to be 0.5 * max(combined_acf).
    (d) width: the minimum width of the peak. It is set to be 2.
(2) The reaosn for the choice is based on the given data. The height is set to be range between 0.01 and 0.5 of the max(combined_acf) because the peaks are usually higher than the rest of the data. The distance is set to be 10 because the peaks are usually far apart. The prominence is set to be 0.5 * max(combined_acf) because the peaks are usually prominent. The width is set to be 2 because the peaks are usually wide.
(3) Some finetuning has been done to the parameters to get the best-possible result, according to the data and ground truth in the train set and test set.

4 TASK P-4

Approaches to detect presence:

(1) Count the motion level or the gains from the ms_bar. If the motion level or the gains are higher than the threshold 0.05, then 1 to the high_motion_count.
(2) If the high_motion_count is more than the 30% of the total number of frames, then classify it as with high motion rate, ie has_high_motion = 1.
(3) If there is high motion rate, and the detected average breathing rate falls in range of 12bpm to 60 bpm, then classify it as presence = 1, else presence = 0.
(4) For this part, I have also tried another method mentioned in the VeCare paper, which is to use the peaks detected in the combined_acf to calculate the breathing rate. However, this method does not work well in the test set, so I have decided to use the method mentioned above
(5) Some idea of this part is inspired by the WiCPD paper [1].

REFERENCES

[1] Xiaolu Zeng, Beibei Wang, Chenshu Wu, Sai Deepika Regani, and K. J. Ray Liu. 2022. WiCPD: Wireless Child Presence Detection System for Smart Cars. *IEEE Internet of Things Journal* 9, 24 (2022), 24866–24881. https://doi.org/10.1109/JIOT.2022.3194873