Using Radar and Neural Networks for Piggybacking detection

Joel Sigurdsson and Hannes Olsson

Faculty of mathematics, LTH

Piggybacking is when a potentially malicious person sneaks behind a person with access through a secure door. We created a system which can automatically detect this with almost 100% accuracy using machine learning. It works in both real-time and for recorded data, and can make decisions on both single images and entire sequences.

While image recognition is a popular field of research, doing so for 3D point clouds, such as the output of a radar, is not as common. For us it posed unique challenges, on top of being really hard to make sense of with the human eye. There already exists camera based technology for detecting piggybacking. We shall evaluate radar as an alternative solution. It has several advantages compared to camera technology, even when not considering technical advantages such as producing three dimensional images. One can not identify a person using radar, therefore radar does not violate integrity in the same way a camera does. On top of this, producing a radar good enough for piggybacking detection is cheaper than producing a camera for the same purpose.

The essence of the thesis was to compare our radar solution to an existing camera solution developed by Axis. Additionally two different placements of the radar were compared, at the side of the door and above the door. The result was that our radar based solution performed slightly better than the camera based solution on easier scenarios, such as one or two people walking normally through a door. For these, both radar positions had accuracies over 99%. When considering more difficult scenarios, such as carrying a large box in front of you, the radars outperformed the camera significantly. When comparing the two placements of the radar, the placement above the door came out slightly ahead. From that point of view, it is easier to see whether one or two people are walking in frame.

A real world scenario where our radar based solution could be useful would be somewhere where security is not of the highest priority. It could for example be used at a subway station to detect when more than one person passes through the gates at the same time. Before deploying it at a subway station, something would have to be done about potential problems that commonly occur at the subway. Such problems could be people traveling with a suitcase or people traveling with a stroller. These are things that could be falsely detected as piggybacking, if not accounted for.

During development a lot of time was spent evaluating different neural networks to find the best possible structure for the piggybacking problem. In the end a type of recurrent neural network called LSTM performed best and was used in the final version. If you are interested in the more technical and pragmatic work of building an as good as possible model for detecting piggybacking, read chapter 3 of the full thesis.