People counter using thermal “tripwire” sensor
Janis Intoy, Emily Lam
EC520: Image Processing and Communications
March 22nd, 2017

Problem Summary

In modern efficient buildings, controlling the HVAC system preemptively can be a huge cost saver. Therefore, a senior design team at Boston University is working towards developing sensing technology and algorithms to count the number of people in a room so that the system is always conscious of room occupancy and can adjust system parameters, such as ventilation, accordingly before feedback sensing technology can detect abnormalities, such as rising temperatures of a crowded room. There are a number of ways to detect occupancy. However, this project assumes that if a room has a low number of entry points, counting people can be done at the entries based on who is entering or leaving the room. As such, there would be a continuous knowledge of the number of people in a room. The senior design team has implemented a privacy-inclined low resolution thermal sensor with a field of view of 30°x120° positioned at the top of the door frame and looking perpendicularly down. It is capable of capturing a 16x4 pixel array at frame rates of 0.5 to 512 Hz. Using this information, our project is tasked at developing algorithms to count the number of people entering or leaving a room. Namely, we will be looking into defining features for data training, testing, and classifying. Good performance would be characterized by a low number of false detections and missed detections.

Review of Literature

A simple way to detect motion or change in an image is to compare a pixel value with a background model, a representation of a scene with no temporal dynamics. Pixels that deviate from the background model are assumed to indicate motion. Kalman filters and Wiener filters are commonly used to model simple backgrounds and are advantageous for backgrounds that change slowly in time [2]. Kalman filters use a series of previous measurements to estimate a probability distribution function of the variable for each time frame; therefore, an unlikely measurement is an indication of a sharp change from the baseline. A Wiener filter is an adaptive low-pass filter that uses local spatial and temporal statistics around each pixel to remove variability of the background from the measurements.

Machine learning methods can further discriminate types of motion based on training data.

K-nearest neighbors (kNN) is is a memory-based, two-stage machine learning algorithm [3-4]. Stage 1 consists of training the model with a labeled data set. Stage 2 classifies test data based on labels of the k-nearest neighbors. A nearest neighbor is defined by a distance, which is typically the euclidean distance. In other words, kNN is a majority vote among the labels of the k nearest training points of a training set.

Support vector machines (SVMs) are linear, binary classifiers that are trained on labeled data sets; however, kernels can be used to map features into a space with higher dimensionality for non-linear classifications. SVM can also be applied in sequence to distinguish between more than two classes. This method uses training data to find the hyperplane that separates the two classes of data in terms of a minimization problem. Incoming data is classified based on which side of the hyperplane it falls into.

Convolutional neural networks (CNNs) is a deep learning technique that is well suited for image classification. They are designed to use minimal amounts of preprocessing, as different layers of the CNN can account for background models and feature extraction [2]. However, training CNNs requires a large amount of data.

Proposed Solution

Some assumptions we are considering when choosing features:

- When entering a room, top portion of the array initially senses warmth while bottom section doesn't and vice-versa for leaving. Therefore, direction can be determined from max value of each row (Figure 1).
- The ambient background frame is slowly changing; there are no drastic changes due to background.

Target Plan:

- Deliverable: Fully working algorithm that can be converted into real-time implementation.
- The algorithm will classify motion using background comparison, KNN, SVM, or CNN on complex features such as:
    - Peak temperature in each row at each time
    - Vectorized spatial and windowed temporal components of the video
    - Mean over all pixels in a frame as a baseline for sensing whether there is a person in the frame or not. (The Occusense report shows that there are different durations of heat in the frame for entering and exiting, i.e. from opening doors [1].)
- Background models can be used as preprocessing and combined with the above machine learning techniques.
- The algorithm will be robust to different light conditions and other external conditions.
- The algorithm will be robust to multiple persons in a single frame.

Backup Plan:

- Simple background comparison to detect motion
- KNN and SVM classification based on basic features
- Not real-time
- Deals with one person at a time

Division of Labor

Task	Janis	Emily
Reports and Documentation	Yes	
Brainstorming features and labels	Yes	
Training Data Set	SVM	KNN
Testing Data Set	Yes	Yes
Extra	CNN	
Comparison of features and machine learning techniques	Yes	
Integrating with Senior Design	Generate random fake data for testing live-stream	Write script to keep track of how many people are in the room

References

[1] Bleda, A., Cepeda, M., Kallarackal, K., Bidnichenko, A., and Nurkatov, Y (2016). First Semester Report: Occusense. Boston University, Electrical & Computer Engineering.
[2] Goyette, N., Jodoin, P., Porikli, F., Konrad, J., and Ishwar, P (2014). A novel video dataset for change detection benchmarking. IEEE Transactions on IMage Processing, 23(11):4663-79.
[3] Ishwar, P. EC503: Learning from Data: Lecture 6: Classification: Nearest Neighbor, Boston University, 2016.
[4]Mathworks Documentation, Classification Using Nearest Neighbors, Accessed online, 2017, https://www.mathworks.com/help/stats/classification-using-nearest-neighbors.html.
[5] Mathworks Documentation, Support Vector Machines for Binary Classification, Accessed online, 2017, https://www.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html#bsr5o09
[6] Mathworks Documentation, Convolutional Neural Networks, Accessed online, 2017, https://www.mathworks.com/help/nnet/convolutional-neural-networks.html

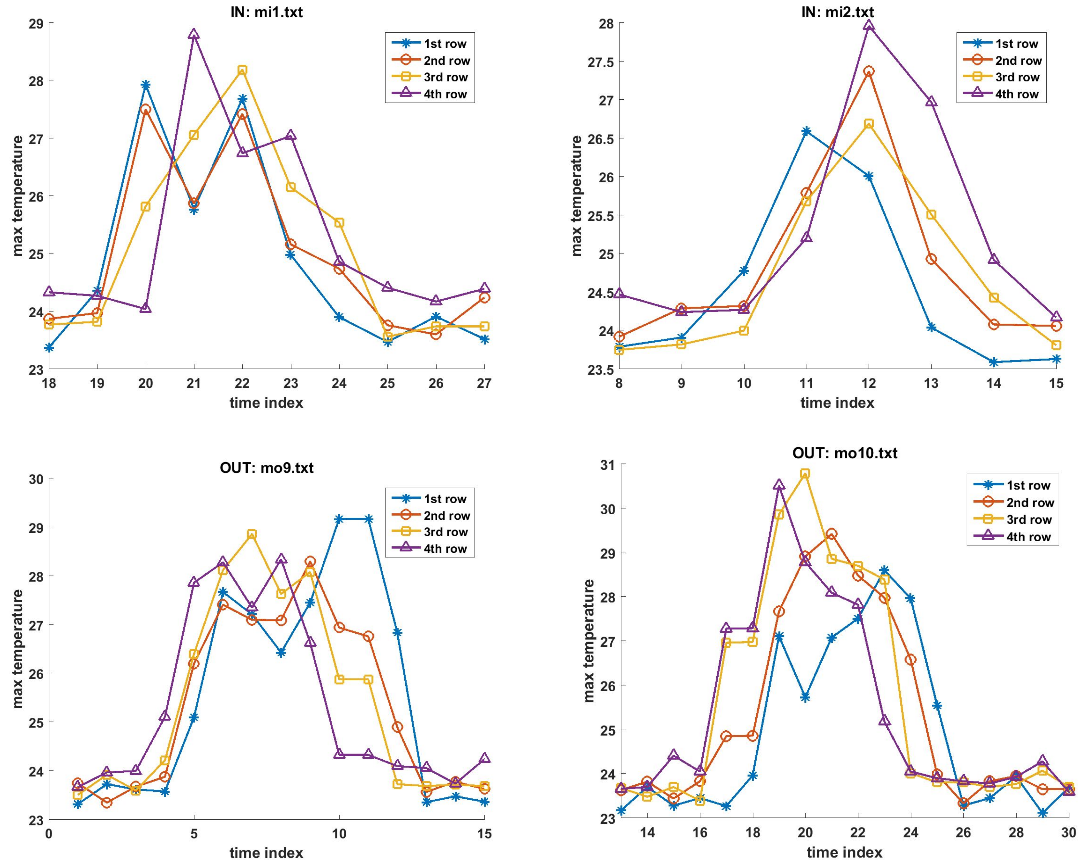

Figure 1: Max temperatures in each row over time in several videos. TOP: Person moving into the room. Row 1 clearly peaks before row 4. BOTTOM: Person moving out of the room: Row 4 peaks before row 1.