# ROOM OCCUPANCY SENSING USING A THERMAL TRIPWIRE

*Janis Intoy and Emily Lam*

May 5, 2017

Boston University
Department of Electrical and Computer Engineering

Technical Report No. ECE-2017-01

**BOSTON UNIVERSITY**

# ROOM OCCUPANCY SENSING USING A THERMAL TRIPWIRE

*Janis Intoy and Emily Lam*

BOSTON UNIVERSITY

Boston University
Department of Electrical and Computer Engineering
8 Saint Mary's Street
Boston, MA 02215
`www.bu.edu/ece`

May 5, 2017

Technical Report No. ECE-2017-01

## Summary

A large part of a building's efficiency depends on the efficiency of its HVAC (Heating Ventilation & Air Conditioning) system, which can be improved with automatic adjustments based on room occupancy level. As thus, accurate prediction of the number of people in a room is needed. In order to a estimate a room's occupancy level, the Occusense Senior Design team has created a privacy-preserving, low resolution, thermal sensor system to capture temperature images of people walking through the doorways. This project uses that information coupled with a background prediction algorithm and optical flow analysis to predict the direction of motion of the people passing through and keep a continuous count of the number of people in the room. Final results are . . . .

# Contents

# List of Figures

# List of Tables

# 1 Introduction

In modern efficient buildings, controlling the HVAC system preemptively can be a huge cost saver. Therefore, Occusense, a senior design team at Boston University, is working towards developing sensing technology and algorithms to continuously keep count of the number of people in a room. This way, the system can adjust system parameters, such as ventilation, accordingly before feedback sensing technology can detect abnormalities, such as rising temperatures in a crowded room. There are a number of ways to detect occupancy, such as using a fisheye camera to count the number of people in a frame. However, this project assumes a tripwire methodology, that is 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, continuous knowledge of the number of people in a room is achieved as a running tally. The senior design team has implemented a 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 of temperatures at frame rates of 8-12Hz. Using this information, our project develops an algorithm to count the number of people entering or leaving a room. Specifically, this algorithm will 1) detect the presence of a moving person in the frame, 2) determine the direction of motion of the person, and 3) keep count of the total number of people in a room.

# 2 Literature Review

Many background subtraction algorithms have been developed to detect changing pixels in a series of images. A brief review of simple background subtraction methods can be found in [1] and more common change detection algorithms in [2]. The most successful amongst these incorporate a background model into the algorithm. The use of a model allows for thresholding of probabilities instead of pixel intensities and is therefore more robust to some variation in the background scene. In addition, foreground models can improve the sensitivity of the change detectors [3]. McHugh et al. suggested a foreground model algorithm that is more general as it is based on spatial neighborhoods. To further improve the discrimination, they also suggest a Markov model so that labels are more spatially coherent [4].

We also looked into algorithms for determining optical flow. Optical flow describes the direction elements in an image move or change with time from frame to frame, usually in a vector form. There are three overarching approaches for determining optical flow: a feature-based approach, a correlation-based approach, and a gradient-based approach. For our project we consider the gradient-based approach as we use background subtraction to determine foreground elements ahead of time. The foreground elements can then be put through a gradient-based optical flow algorithm to determine the motion of the object. This can be modeled as a gradient with partial derivatives spatially and temporally for each pixel intensity, as each pixel will have a change in both the x-direction, y-direction, and time [5].

# 3 Problem Statement

Our approach breaks the problem into three parts: 1) detect a person in the doorway, 2) determine the direction of motion of the person, 3) classify the motion and track total number of people in a room.

## 3.1 Person Detection

In order to accomplish this we make several assumptions about the acquired data. First, we assume that the background has slow temporal dynamics. Second, we assume that the data starts with some number of background-only frames. For business and residential buildings this is likely true if they close for the night. Finally, we assume that the surface temperature of a single body as detected by the sensor has the same variance as the background and that it will be greater than the temperature of the background. Here we propose two background subtractions method to detect a person or persons in the frame. One method to detect foreground in each frame is to model the background temperatures in each pixel as a gaussian which is updated with each new frame. The parameters of the background model are updated when the current pixel is labeled as background:

$$\mu_t = \rho I_t + (1-\rho)\mu_{t-1}$$

$$\sigma_t^2 = (I_t - \mu_t)^2\rho + (1-\rho)\sigma_{t-1}^2$$

where $I_t$ is the pixel temperature and $\mu_t$ and $\sigma_t^2$ are the mean and variance of the gaussian in frame $t$. A Markov model can be applied to improve the spatial coherency of foreground labels: a pixel surrounded by foreground pixels has a low threshold since it is likely also foreground. Therefore, a pixel is labelled as foreground when

$$\frac{I_t - \mu_t}{\sigma_t} >_{\mathcal{F}} \theta \exp((Q_{\mathcal{B}} - Q_{\mathcal{F}})/\gamma)$$

where $\theta$ is an initial threshold (# of standard deviations), $Q_{\mathcal{B}}$ and $Q_{\mathcal{F}}$ are the number of neighboring background and foreground pixels respectively, and $\gamma$ is a parameter that changes the influence of the Markov model on the threshold.

The second proposed solution is inspired primarily by the foreground-adaptive background subtraction described in [4]. The background can be modeled by a probability density function estimated at each pixel location $\mathbf{n}$ from the $N$ most recent pixels that were labeled as background:

$$P_{\mathcal{B}}\left(I^{(k)}[\mathbf{n}]\right) = \frac{1}{N}\sum_{i\in\mathcal{B}_k[\mathbf{n}]} \mathcal{K}\left(I^{(k)}[\mathbf{n}] - I^{(i)}[\mathbf{n}]\right)$$

where $I^{(k)}[\mathbf{n}]$ is the temperature at location $\mathbf{n}$, $B_k[\mathbf{n}]$ are the $N$ previous time indices at which the pixel located at $\mathbf{n}$ was labeled background, and $\mathcal{K}$ is a zero-mean Gaussian with variance $\sigma^2$. Thresholding these probabilities with threshold $\theta$ gives an initial classification of pixels into background or foreground.

The development of the foreground model follows a similar formulation, but the summation is over pixels in the neighborhood of $\mathbf{n}$ that have been already been labeled as foreground. The ratio of the probability density functions $P_{\mathcal{B}}$ and $P_{\mathcal{F}}$ can be thresholded to determine new labels for each pixel. Furthermore, the number of foreground neighbors a pixel has can be used to adjust the threshold. The more foreground neighbors a pixel has the easier it is to be labeled as foreground. This will contribute to the spatial coherency of the labels. Thus, a pixel is labeled as background if

$$\frac{P_{\mathcal{B}}(I[\mathbf{n}])}{P_{\mathcal{F}}(I[\mathbf{n}])} > \theta \exp\left(\frac{1}{\gamma}(Q_{\mathcal{F}}[\mathbf{n}] - Q_{\mathcal{B}}[\mathbf{n}])\right)$$

where $Q_{\mathcal{F}}[\mathbf{n}]$ and $Q_{\mathcal{B}}[\mathbf{n}]$ are the number of neighbors that are foreground and background respectively and $\gamma$ is a parameter to control how strongly the threshold adapts. The thresholding step can be done iteratively as labels are updated in each step.

These algorithms result in labels for each pixel in each frame as background or foreground. The number of foreground pixels in each frame could indicate whether a person is in the frame.

## 3.2 Direction Discrimination

After determining the foreground from the background, we look at determining the direction of flow of the foreground elements. We do this by taking the foreground elements only, setting the background elements to zero, and applying a gradient-based optical flow algorithm. This can be modeled as described in [5]:

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0$$

Since we are only concerned about the vertical velocity, i.e. a person walks through the door frame and the laterally motion is trivial, the previous equation can be written:

$$v_y = -\frac{\frac{\partial I}{\partial t}}{\frac{\partial I}{\partial y}}$$

where $v_y = \frac{\partial y}{\partial t}$. A large $v_y$ would correspond to a lot of motion in that direction. In our system of 4x16 array of pixels, we generate an optical flow matrix consisting of 3x16 pixels to take the boundary condition into account. Next, we perform two averages, we average the rows to find the optical flow of the row, and we then average those row averages together to obtain an overall optical flow for the entire array. This overall velocity value is used to determine the optical flow of the foreground which determine whether a person is walking into or out of the room.

# 4 Implementation

The current implementation of the algorithm is in Matlab. The three parts of the algorithm (Fig ...) are implemented separately and can be run in sequence to output a single room occupancy number.

## 4.1 Data Acquisition

In addition to the labelled data from the senior design team, we acquired several trials of realistic but more difficult situations. This includes a person lingering in the doorway and multiple people passing through the door in quick succession.

The data is recorded into text files at a rate of 8-12Hz. The data we currently have was recorded at 8Hz though the Occusense team has updated the sampling rate to 12Hz.

## 4.2 Person Detection

Janis implemented both proposed background subtraction algorithms in backgroundSubtractionSimple.m (running gaussian) and backgroundSubtraction.m (kernel-based approach). They take as input a structure of parameters and a 3-D array of temperatures over the two spatial and one temporal dimensions. This implementation could be modified to run in real time and, depending on the size of the history for the computation of the background PDF, would not require too much memory.

## 4.3 Direction Discrimination

Emily implemented this algorithm in the file opticalflow.m. It takes as input a 3-D array of temperatures over the two spatial and one time dimensions. This can be the raw information provided by the sensors or a masked foreground-only version that has already undergone background subtraction. This implementation could be modified to run in real time to follow the background subtraction function.

## 4.4 Direction Discrimination

Janis and Emily implemented this algorithm in the file pCounter.m. This files takes the output of the opticalflow.m, a series of vertical gradients, and averages them per a frame and calculates the temporal gradients based on . . .

# 5 Experimental Results

Algorithms were tested on simulated data generated by randomly concatenating the labelled data. The temperatures of the frames are shifted for continuity between frames.

### 5.1 Background Subtraction

The background subtraction results are shown in the following ROC curves.The kernel-based method had overall better performance than the parametric method achieving hit rates of over 90% for the best parameter sets with negligible false alarms in detecting a person in the frames. Most misses occurred because of spontaneous rapid changes in temperature of background pixels. These occur infrequently in the data but also arise when true foreground pixels are mislabelled and bias the background model.

The best parameters for the kernel-based method had strong influence from the Markov random field $0.2 \leq \gamma \leq 0.8$ based on a first order neighborhood, and a gaussian kernel with $\sigma = 0.5$ (approximately the standard deviation of the background estimated from the data).

### 5.2 Direction Discrimination

This method seems to work well in detecting direction. However, it is still not accurate all the time, $\sim 80\%$. In Figure **??**., we threshold to see if the velocity is large enough to count as a person moving in a direction as there are some bounce back residual effects. For some cases, it detects the person as moving in the opposite direction.

## 6 Conclusions

- Large-scale testing: We can simulate more and longer test videos to measure performance of the algorithms and their different parameter varieties. Based on these results we can modify the algorithms to improve performance.
- Data Acquisition: We will collect more difficult data to test the robustness of the algorithms.

## 7 Figures

# References

[1] M. Piccardi. "Background subtraction techniques: a review," *IEEE International Conference on Systems, Man and Cybernetics*, 2004.

[2] N. Goyette, P. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, "A Novel Video Dataset for Change Detection Benchmarking," *IEEE Transactions on Image Processing*, vol. 23, pp. 4663-79, Nov. 2014.

[3] A. Elgammal, R. Duraiswami, D. Harwood, and L. Davis, "Background and Foreground Modeling Using Nonparametric Kernel Density Estimation for Visual Surveillance," *Proceeding of the IEEE*, vol. 90, pp. 390-393, July 2002.

[4] J. McHugh, J. Konrad, V. Saligrama, and P. Jodoin, "Foreground-Adaptive Background Subtraction," *IEEE Signal Processing Letters*, vol. 16, pp. 390-393, May 2009.

[5] S. Smith, "Reviews of Optic Flow, Motion Segmentation, Edge finding and Corner Finding," *Technical Report TR97SMS1, Oxford Centre for Functional Magnetic Resonance Imaging of the Brain (FMRIB)*, https://users.fmrib.ox.ac.uk/ steve/review/review/review.html, 1997.