MASTER THESIS

Dynamic HVAC operations based on occupancy patterns with real-time vision-based system

Siliang Lu

Building Performance and Diagnostics

School of Architecture

College of Fine Arts

Carnegie Mellon University

Pittsburgh, Pennsylvania, United States

May, 2017

Table of Contents

List of Figures

List of Tables

Abstract

An integrated heating, ventilation and air-conditioning (HVAC) system is one of the most important components to determining the energy consumption of the entire building. For commercial buildings, particularly office buildings and schools, the heating and cooling loads are largely dependent on the occupant behavioral patterns such as occupancy rates and their activities. Therefore, if HVAC systems can respond to dynamic occupancy profiles, there is a large potential to reduce energy consumption. However, currently, most of existing HVAC systems operate without the ability to adjust supply air rate accordingly in response to the dynamic profiles of occupants. Due to this inefficiency, much of the HVAC energy use is wasted, particularly when the conditioned spaces are unoccupied or under-occupied (less occupants than the intended design). The solution to this inefficiency is to control HVAC system based on dynamic occupant profiles. Motivated by this, the research provides a real-time vision-based occupant pattern recognition system for occupancy counting as well as activity level classification. The proposed vision-based system is integrated into the existing HVAC simulation model of a U.S. office building to investigate the level of energy savings as well as thermal comfort improvement compared to traditional existing HVAC control system.

The research is divided into two parts. The first part is to use an open source library based on neural network for real-time occupant counting and background subtraction method for activity level classification with a common static RGB camera. The second part utilizes a DOE reference office building model with customized dynamic occupancy schedule, including the number of occupant schedule, activity schedule and clothing insulation schedule to identify the potential energy savings compared with conventional HVAC control system.

The research results revealed that vision-based systems can detect occupants and classify activity level in real time with accuracy around 90% when there are not many occlusions. Additionally, the dynamic occupant schedules indeed can bring about energy savings. Details of vision-based system, methodology, simulation configurations and results will be presented in the paper as well as potential opportunities for use throughout multiple types of commercial buildings, specifically focused on office and educational institutes.

Key Results in this thesis include:

- Validation of vision-based occupant recognition system
- Energy simulation to see the potential of energy savings and comfort level improvements with dynamic occupancy schedule
- Complete result

Acknowledgement

I would like to thank everyone who I have helped me in my studies for the past two years. For my thesis, I especially want to thank the following people who are either my respectful professors or my friends.

- Thanks **Prof. Erica & Azizan** for providing insightful suggestions for my project every week
- Thanks **Sirajum Munir & Charles Shelton** from Bosch Pittsburgh for developing such wonderful occupancy estimation tool of FORK
- Thanks **Anand Krishnan** from ESTP for helping me set up the experiment in Scaife Hall 212
- Thanks **Yuan Yuan** from RI for introducing Darknet to me
- Thanks **George Wang** from CD for allowing me to use his GTX 970 for my project
- Thanks **Sida Wang** from RI for allowing me to use her own laptop for flashing the device
- Thanks **Luna Zhang** from ETC for allowing me to use her desktop to test the performance with her GPU
- Thanks **Jin Hu** from INI for answering many basic questions and helping me establish the framework of EE
- Thanks **all my classmates from BPD**, especially **Yuqi Pan** for meaningful discussions with me almost every day
- Thanks **all participants** in my three experiments
- Thanks **Xuan Luo** from BPD for providing three wonderful papers and helping understand occupancy simulator
- Thanks **Prof. Vivian & Volker** for helping me get through all my work before thesis presentation
- Thanks **Prof. Nina** for helping me understand the energy systems and new applications with fuel cell during last summer
- Thanks **all my teammates** who have helped me a lot in other courses

Last but not least, I want to say thank you to my family, especially my parents **Qinan Lu** (my father) and **Mi Zhou** (my mother) who are always supporting me whatever decisions I have made and believing in me throughout my whole life. Thanks a lot!

1. Introduction

1.1 Motivation:

An integrated heating, ventilation and air-conditioning (HVAC) system is one of the most important components to determining the energy consumption of the entire building. Currently, the amount of energy used for HVAC has accounted for 37% of all energy used in commercial buildings (EIA, 2012) across the United States. Meanwhile, HVAC is also a key to providing great indoor environment quality (IEQ) and maintaining human comfort, including indoor air quality (IAQ), indoor thermal comfort as well as indoor acoustic comfort. Since HVAC system is responsible for removing the extra heating and cooling loads effectively, it is believed that the optimal HVAC system shall have the ability to adjust the quantity of supply air according to what is needed in real time. For most of the commercial buildings, particularly office buildings and schools, the heating and cooling loads are largely dependent on the occupancy behavioral patterns like presence and activities. Therefore, if HVAC can be responsive to the dynamic occupancy profile, it has a large potential to reduce energy consumption. However, currently, most of the existing HVAC systems have been operated without the ability to adjust supply air rate accordingly. Therefore, much of the energy use for HVAC is wasted, particularly when the conditioned spaces being unoccupied or the operation being under the maximum levels.

Moreover, in order to develop a reliable, dynamic HVAC system control, it is necessary for the system to get the occupancy information on time or get the occupants involved. Motivated by this, many researchers have studied various approaches to advanced HVAC design by integrating occupancy behavioral patterns into control system. Currently, the approaches can be categorized into direct or indirect ways to develop occupancy recognition systems. Compared to indirect approaches, which use environmental sensor data mining, energy consumption data mining, and other stochastic modeling approaches(Zhao, 2015), the direct approaches are less investigated, which use technique directly to get the information of occupancy like the number of occupants indoors. Even if the direct approaches may have the risk of privacy, it is less complex and easier to be implemented than indirect approaches.

Moreover, with the rapid development of artificial intelligence, particularly computer vision, the direct approaches to developing dynamic occupant pattern recognition system has drawn much more attention than before. By implementing occupant detection, tracking and recognition system, it is feasible to construct a real-time occupant profile so as to control HVAC system dynamically by installing multiple cameras in the building instead of a large scale sensor network.

Hence, the project is aimed to propose a vision-based occupant pattern recognition system with a static RGB camera based on an open-source deep learning algorithm and integrated dynamic occupant profiles generated with Occupancy Simulator into the existing HVAC control system in an office building to see its potential energy savings as well as with Energyplus.

1.2 Deliverables:

- A validated real-time vision-based system with a static RGB camera for occupancy detection, occupancy counting and occupancy activity classification based on an open-source deep learning algorithm;
- An energy simulation with three sets of dynamic HVAC operation strategies to see the potential benefits of occupant-oriented HVAC control based on dynamic occupancy profiles generated from Occupancy Simulator;
- A complete report including the validation of the vision-based system as well as the comparisons of simulated energy performance three sets of HVAC operational strategies;

1.3 Hypothesis:

A real-time vision-based system which recognizes occupant behaviors will increase HVAC energy efficiency and improve occupant comfort level.

- Vision-based system can be utilized to detect the occupant and recognize three different actions: walking, standing, and sitting;
- Vision-based system can provide dynamic occupant information as an input to the HVAC control system;
- Dynamic HVAC control based on a vision-based system will save more energy and improve occupant comfort level compared to a conventional HVAC control with fixed schedules.

2 Background of dynamic HVAC operations with occupancy patterns

Based on recent market surveys, HVAC control is believed to bring about potential energy savings varying from 5% to 20% (Dong, 2010). In order to develop an optimal HVAC control, the quantity of supply air shall be adjusted according to what is needed, which can be seen as dynamic HVAC control. Occupancy is one of the most important factors affecting HVAC performances in most commercial and residential buildings as it has a significant influence on the heating and cooling loads. The conventional HVAC control utilizes fixed schedule based on the maximum occupancy assumptions. Since dynamic HVAC control has appreciably great energy savings, studies on how to detect the presence of occupants and predict the occupant behavioral patterns so has to construct dynamic profiles/schedules have played an import role in research field of sustainable buildings and promote the progress in HVAC industry.

2.1 Importance of occupancy patterns to HVAC operations

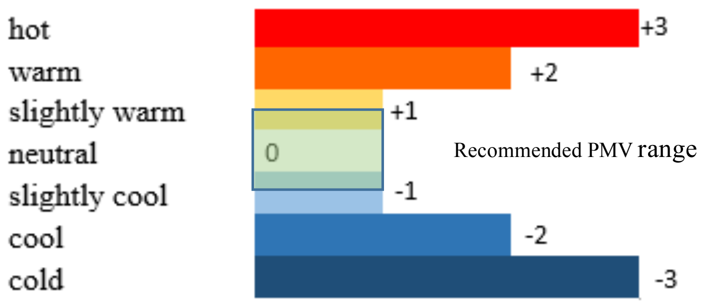
Being informed of real-time conditions in different thermal zones of a building is important to HVAC energy performance. In order to reduce energy consumption of the building, HVAC operational strategies have to be different for occupied and unoccupied zones during different periods of the day. Moreover, since for HVAC system, cooling or heating the air and supplying enough amount of the airflow consume the energy mostly, in order to minimize the energy consumption, it is significant to set appropriate supply temperature and appropriate airflow rate into the zones. According to ASHRAE standards, the room set points and supply airflow rate are both closely related to the occupancy due to heating, cooling or ventilating requirements during heating, cooling and transitional seasons. Kaiyu Sun & Tianzheng Hong (Sun & Hong, 2017) studied potential energy savings based on simulation approach where it was found that the occupant behavior measures can achieve overall site energy savings as high as 22.9% for individual measures and up to 41% for integrated measures.

In addition, Jie provided a comprehensive discussion on the influence of occupant behaviors on the HVAC system(Zhao, 2015). He summarized both the “active” and “passive” role of occupants in the building. For “active” role, it showed that 10 out of 15 studies demonstrated that occupant’s “active” role have an energy impact by using various individualized control systems and dashboards for building HVAC, lighting and plug load systems. For “passive” role, it showed that the occupant’s “passive” role also impacts the building performance, including thermal and indoor air quality, HVAC energy consumption. Moreover, he pointed out it is critical to study the occupant behavior at both individual and group levels and numerous methods have been developed to study them, which will be discussed in **section 2.3.**

Therefore, if real-time occupancy information is available, such as presence of occupants, occupant density, occupant activities (e.g. standing, sitting), not only more appropriate set points and airflow rate for each zone can be determined so as to reduce energy consumption but also better indoor environment quality can be achieved.

2.2 Importance of HVAC operations to occupants

As mentioned before, indoor environmental quality (IEQ) is a key factor to occupants’ productivity and health since people spend approximately 90% of their time indoors (EPA,2017). Among these qualities, which mainly include visual quality, thermal quality, air quality and acoustic quality, thermal and air quality are highly related to the operation of HVAC system. In addition, many researchers have been investigating occupant comfort through various methods, including measurement and modeling. One of them is the predicted mean vote (PMV) model. PMV uses heat balance principles to relate the six key factors for thermal comfort, which are clothing insulation[clo], metabolic rate[met], external work (normally around 0), air temperature[C], mean radiant temperature[C], relative humidity [%] and relative air velocity[m/s], to the average response of people on the thermal sensation scale defined as follows (ASHRAE 55):



Since HVAC operation strongly influences the indoor air quality and thermal quality in terms of indoor particle concentration, temperature and humidity, it is of great significance to have a systematic operational strategy to control HVAC system to ensure comfortable and healthy indoor environment for occupants. Therefore, HVAC operational strategies not only have to take energy consumption of the whole building into account but also has to consider individual's thermal comfort as well as indoor air quality.

2.3 Approaches to studying occupancy patterns in the buildings

As mentioned before, a plenty of methods have been developed to study occupant behavior at both individual and group level(Zhao, 2015), which were categorized as indirect approaches and direct approaches. Meanwhile, some researchers from USC have categorized the occupancy pattern methods with different terms, which are individualized and non-individualized methods(Li, Calis, & Becerik-Gerber, 2012). Since the definitions of these different categories, non-individualized approach is similar to indirect approach and individualized approach is similar to direct approach, in order not to confuse the readers, only indirect approaches and direct approaches to occupant pattern recognition are discussed here.

2.3.1 Indirect approaches to occupant pattern recognition

According to Zhao (Zhao, 2015), the indirect approaches are less intrusive occupant positioning and recognition methods, which use environmental sensor data mining, energy consumption data mining, and other stochastic modeling approaches. Such methods integrate simulations, experiments in order to better control HVAC without little or even no individual information about the occupants. Therefore, in this section, both simulation-based approaches and sensor-based approaches are discussed.

➢ Simulation-based approach

The Lawrence Berkeley National Laboratory (LBNL) has developed an agent-based occupancy simulator for building performance modeling(Chen, Luo, & Hong, 2016)(Luo et al., 2017). The web-based application simulates each occupants as an agent with specific movement events and statistics of space uses. Meanwhile, the simulator simulates the location of each occupant at each time step based on the first-order homogeneous Markov chain model. The structure of the occupancy simulator is shown in Figure 2.

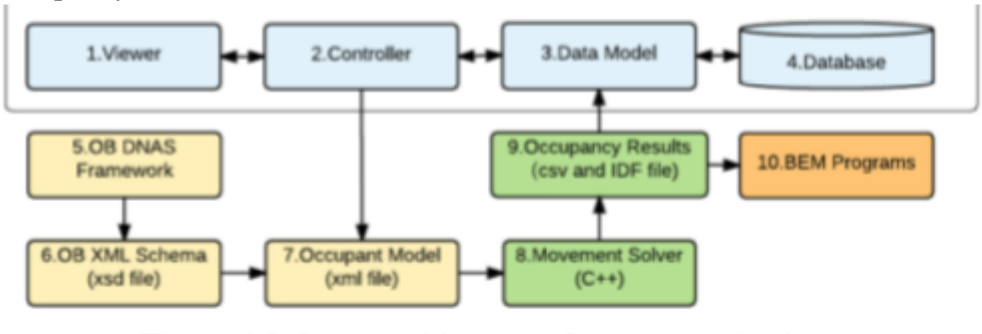


Figure 2 Agent-based occupancy simulator

With such interactive occupant schedules, the simulations conducted from EnergyPlus and obFMU can more accurately evaluate the impacts of occupant behaviors on building energy performance. The simulated results of the case study showed that with the proposed simulator, the lighting consumption could achieve significant reduction.

Moreover, Zhao has developed and demonstrated the concept of Design-build-operate Energy Information Modeling (DBO-EIM) infrastructure, which involves occupant-oriented predictive building control for reducing energy consumption and maintaining occupant thermal comfort at the same time (Zhao, 2015). Figure 3 shows the concept of "Receding horizon" for the integration of EnergyPlus and Predictive control.

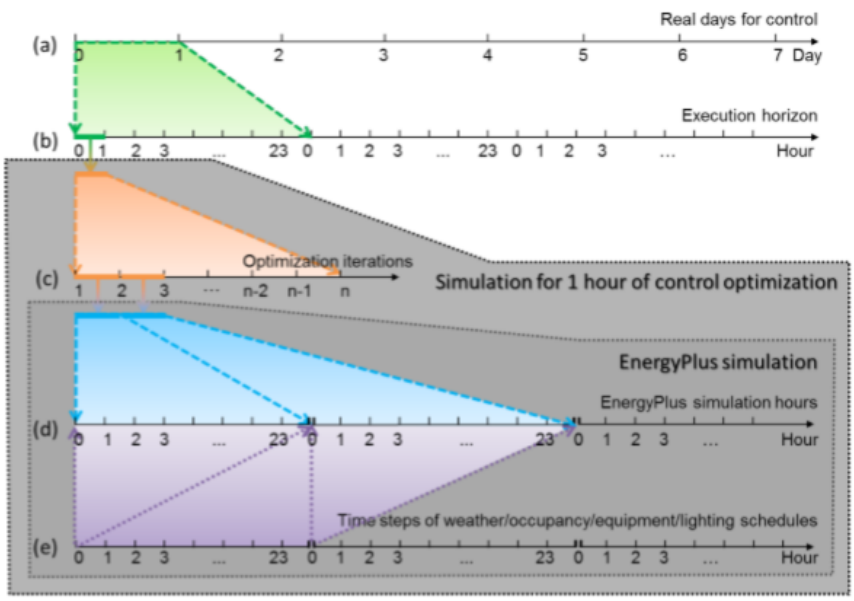


Figure 3 Receding horizon for the E+ and predictive control

- Sensor-based approach

As mentioned before, the indirect approaches can also be implemented by collecting information related to occupancy so as to predict the presence of occupancy and the occupant behaviors. Besides Zhao, another researcher from CMU set up a large-scale sensor network so as to implement a nonlinear model predictive control (NMPC) in the test bed (Dong, 2010). Artificial neural network (ANN) and Hidden Markov model (HMM) were implemented to predict the number of occupants with the inputs of CO2 concentration, total volatile organic compounds (TVOC), PM 2.5, acoustics, illumination, motion, temperature, humidity as well as a video camera network. Moreover, the optimization algorithm of dynamic programming was implemented to optimize the NMPC problem which was solved with Matlab for controller design and LabView for actuation. The overview of optimal control implementation schema is shown in Figure 4. The results showed that energy reduction could reach nearly 50% based on this proposed control approach for the entire heating/cooling testing periods.

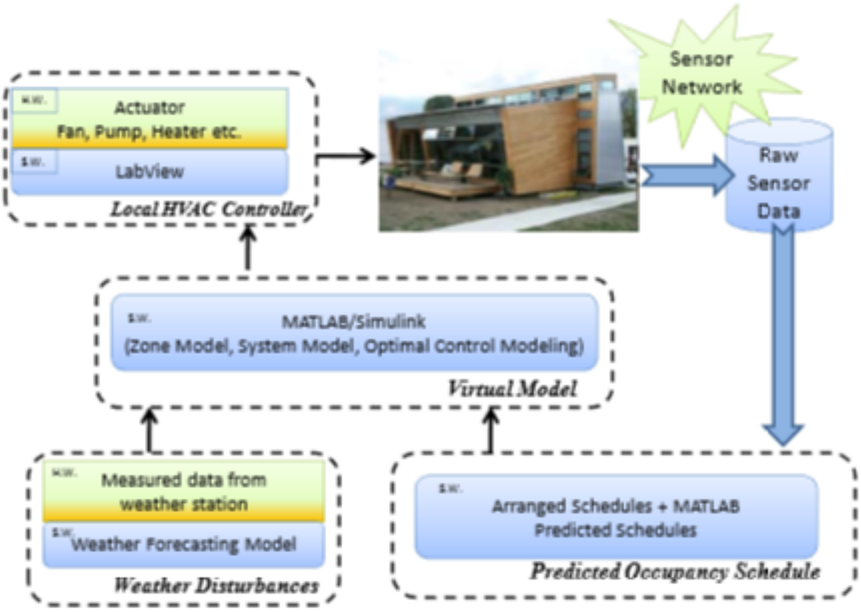


Figure 4 Optimal control implementation schema

2.3.2 Direct approaches to occupant pattern recognition

Even if indirect approaches have achieved energy reduction and improved human comfort level, the accuracy and complexity for real implementation in a building is still not satisfactory. Compared to indirect approaches, direct approaches have a great advantage of higher accuracy in terms of occupancy counting, activity level measurement. Even if direct approaches may have privacy issue, more and more researchers have turned to study the occupant patterns in a direct way. Due to the rapid development of machine intelligence, particularly computer vision and wireless sensor network, the direct approach of developing the vision-based system has drawn much more attention than before, which can be referred as ambient intelligent system.

- Vision-based sensing system

  A group of researchers from France proposed a vision-based system for human detection and activity analysis based on video sequences using a static camera(Benezeth, Laurent, Emile, & Rosenberger, 2011) involved in the *"CAPTHOM"* project. The vision-based system is comprised of background subtraction, tracking, occupant recognition and activity characteristics. The process for human detection and activity characterization in video sequences is shown in Figure 5.

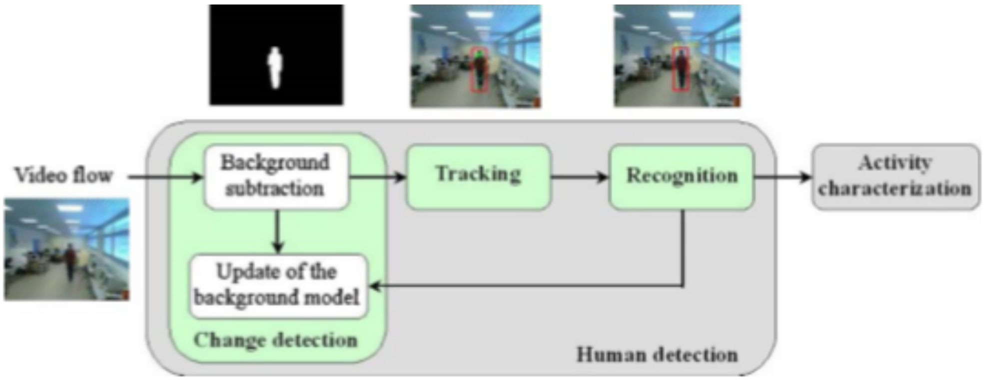


Figure 5 Framework of human detection

For background subtraction, the proposed system utilized a Gaussian probability density function to model each pixel. For tracking the moving objects, the system utilized the Lucas-Kanade tracker. For occupant recognition, the system implemented the algorithm proposed by Viola and Jones, which is based on Haar-like filters and a cascade of boosted classifiers built with several weak classifiers trained with a boosting method and detected humans with a set of bounding boxes. For activity characterization, the proposed system classifies activities as either "Quiet" or "Active" based on the ratio between the number of pixels in motion and the number of pixels in the foreground. The detection result $A_{s,t}$ formulated as:

$$A_{s,t} = \begin{cases} 1 & if\ d2(Is,t - Is,t-\eta) > \tau a \\ 0 & otherwise \end{cases}$$

where $d2(Is,t - Is,t-\eta)$ is the Euclidean distance between the pixel s at time t and the same pixel at time $t-\eta$ and τa is threshold. The experiments were tested in various space types, including offices, meeting rooms, corridors and dining rooms. The results showed that the proposed system achieved a correct detection rate of 97% for occupancy counting and a false detection rate of 3%. In addition, Figure **6** shows the confusion matrix of activity level classification.

	Quiet	Active
Quiet	0.91	0.09
Active	0.00	1.00

Figure 6 Confusion matrix for activity level classification

In addition, Liu et al. also proposed a dynamic Bayesian network-based method based on the video surveillance for occupant detection both at the entrance and inside the room. Three two-stage static detector was presented and compared to find the human heads inside the room and tracking was used for occupant detection at the entrance (Liu et al., 2013) with the fusion of vision sensors.

Instead of using RGB camera, another researcher conducted a continuous 24h experiment of monitoring the building for Type equation here. commissioning purposes(Shih, 2014). The system utilized PTZ cameras, depth cameras to test a robust day-and-night people tracking and counting algorithm. The system consisting of background modeling, comprehensive motion detection, tracking algorithm. The flow diagram of the proposed monitoring system is shown in Figure 7. Unlike Lucas-Kanade tracker, the system proposed a tracking algorithm based on particle filtering and support vector machine (SVM) with online learning.

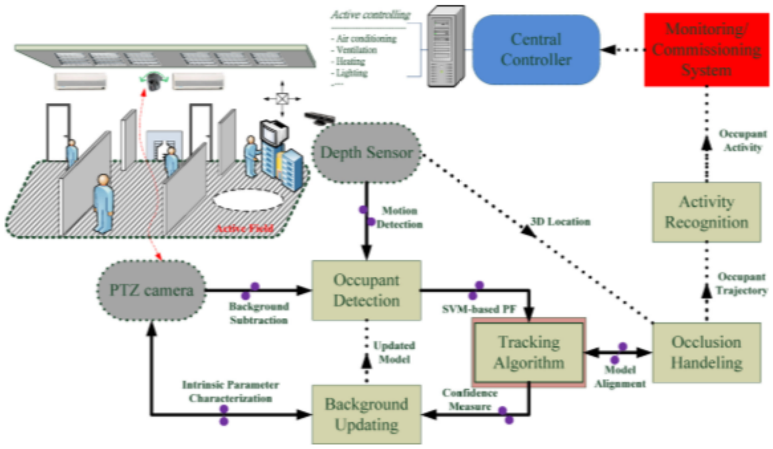


Figure 7 Framework of occupancy recognition using depth sensor

One of the recent occupancy estimation systems has been developed by Bosch Research and Technology Center (Munir et al., 2017). Similar to Shih, they've developed a prototype called FORK using a lightweight computer vision algorithm using depth information to detect occupants. The FORK is mounted at the door so as to monitor people's entrance and exit. Compared to other vision-based approaches, FORK is less private and intrusive and it is expected to achieve over 99% accuracy with 4-9 FPS for a real deployment in a building despite the cost for each unit being as high as $260. Table 1 shows the summary of different state-of-art vision-based occupancy pattern recognition system.

Table 1 existing vision-based occupancy pattern recognition system

	FORK	**Fusion of vision sensors**	**image-based occupancy detection**	**CAPTHOM**
Author	S. Munir et al.	D. Liu et al.	H. Shih	Y. Benezeth et al.
Sensor type	Depth sensor(Kinect)	surveillance camera	Kinect & PTZ	static RGB camera
detection algorithm	Multi-level scanning with depth information	Haar only; Haar+Contour;Haar+HoG	depth background subtraction	Multi-classifiers for different body parts
tracking algorithm	greedy bipartite matching	Background subtraction	Particle filtering	Lucas-Kande
Accuracy for occupancy counting	99%	94%	Not mentioned	97%

- Other direct occupancy sensing system

  Besides vision-based system, other direct approaches include cellular data, ultrasonic technology, wearable devices like Fitbit and Radio-Frequency Identification (RFID) for occupancy recognition system. **Table 2** shows pros and cons of different occupant sensing systems besides vision sensors.

Table 2 Pros and cons of different occupant sensing systems

Sensor	***Pros***	***Cons***
GPS	High Accuracy outdoors	Privacy(Zhao, 2015); less accuracy indoors
Ultrasonic sensor	Easy implementation	low accuracy, noise sensitivity (Nursgejarietal, 2016); significant training is necessary(Shih et al., 2015)
RFID	Easy to implement	Privacy(A Juels, 2014); limited range (Liu, 2014)
Cellular data	Easy to get the information	Privacy(Zhao, 2015)
Wearable devices	Portable (Diraco et al., 2015)	Intrusive (Diraco et al., 2015)

- Direct approaches to thermal comfort

  Most of the direct approaches to occupancy pattern recognition described above are focused on occupancy counting; however, it could be also significant to analyze thermal comfort level of an individual by implementing different types of sensors such as bio-sensors and Infrared (IR) camera.

  - Bio-sensing for adaptive thermal comfort

    Joon Ho Choi established an adaptive thermal comfort control based on real-time human physiological responses or bio-signals such as individual skin temperature to operate a local HVAC system, as shown in Figure 8. Such bio-sensing controllers can be utilized in various types of buildings like office building, healthcare and residential buildings. The results showed that such bio-sensing system significantly increase individual comfort while achieved 1.1% of annual energy savings. However, the work hasn't been extended for multi-occupancy conditions where people share a single HVAC system (Choi, 2010).

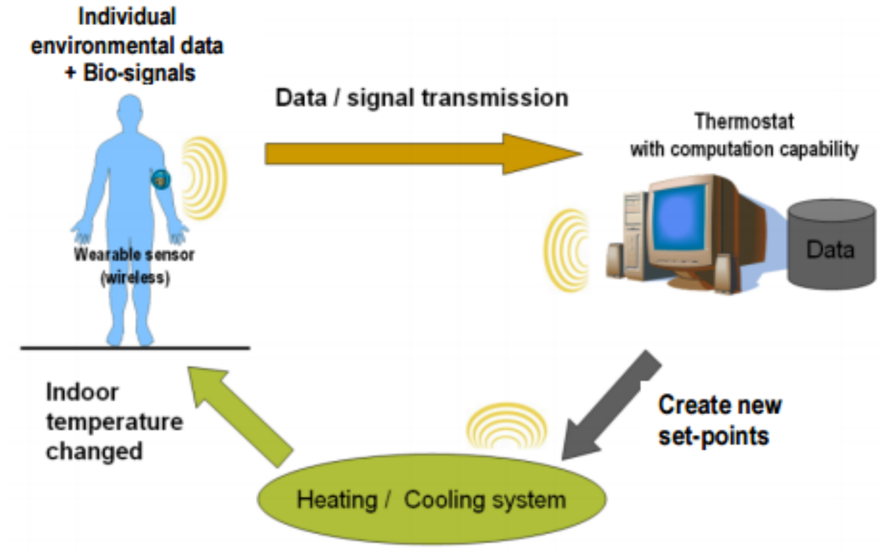


Figure 8 Adaptive thermal comfort control based on real-time human physiological responses or bio-signals

- IR camera for thermal comfort analysis
  Ranjan and James conducted research to see the potential of detecting and predicting dynamic thermal comfort using IR imaging to detect skin temperature and validate the predictions via post-study survey. The results showed that on around 40% of occasions, the HVAC system could have less consumption to achieve comfort. In addition, thermographic imaging has been validated to be an effective tool to infer if HVAC system has to be turned on to maintain occupant comfort with an accuracy of 94-95%.

Based on my literature review, it can be concluded that the recent studies on occupancy pattern recognition technique for dynamic operation of HVAC system has turned to direct approaches, especially vision-based system due to rapid development of computer vision and wireless sensor network. For occupancy pattern recognition system, several major factors have to be under consideration: accuracy, intrusiveness, cost and privacy. Compared to indirect approaches like sensor fusion of different environmental sensors, vision-based occupancy sensing has better accuracy. Meanwhile, compared to direct approaches like bio-sensors or smart phone, vision-based occupancy sensing is not that intrusive. Moreover, among three major vision sensors, which are RGB sensor, depth sensor and IR sensor, RGB sensor is the most cost-effective one. Even if RGB sensor may be faced with privacy issue, this can be solved under different conditions, more details will be provided in **Discussion** section.

Motivated by this, the research proposes another real-time vision-based occupant pattern recognition system for occupancy counting as well as activity level classification with RGB camera based on an open-source neural network library. Moreover, a dynamic occupancy schedule is integrated into the existing HVAC simulation model of a U.S. office building to investigate the level of energy savings as well as thermal comfort improvement with occupant-oriented HVAC operational strategies compared to traditional existing HVAC operational strategies.

3 Methodology

The methodology can be divided into four parts, which are literature review, development of real-time occupancy pattern recognition system, simulation of a DOE reference building to see the potential of energy savings and comfort improvement based on dynamic occupant-oriented HVAC operational strategies, evaluation of the proposed occupancy pattern recognition system and result analysis of the energy simulation.

1) **<u>Literature Review:</u>**

The literature review covers the research progress related to the project, which can be categorized as follows:

a) Studies on importance of occupancy patterns to HVAC operations
b) Studies on importance of HVAC operations to occupants
c) Studies on indirect approaches to developing dynamic occupant schedules, including simulation-based method and sensor-based method;
d) Studies on direct approaches to developing dynamic occupant schedules, including vision-based occupant recognition system and other recognition systems in terms of energy consumption and thermal comfort;

2) **<u>Development of occupancy pattern recognition system:</u>**

The HVAC operation pipeline is shown in Figure 9:

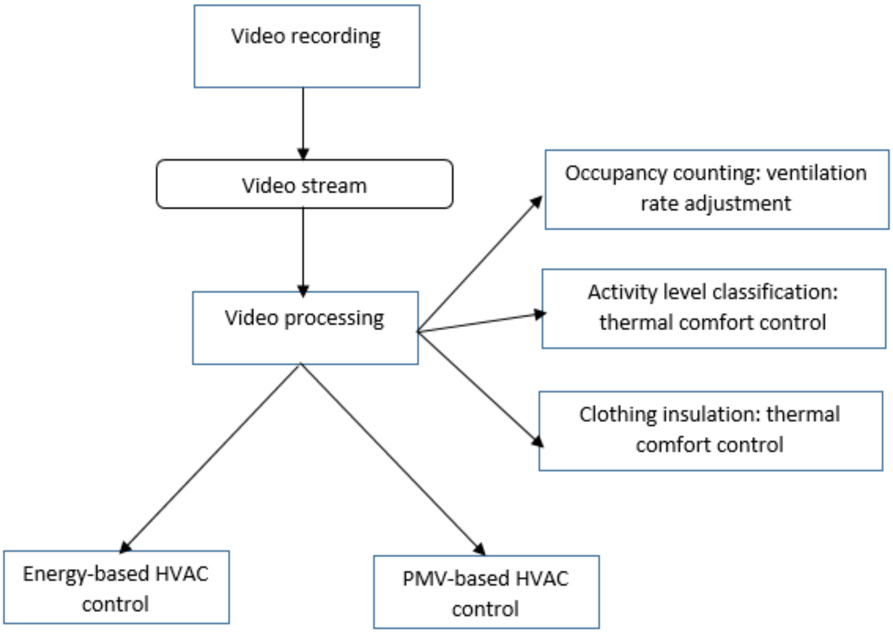


Figure 9 HVAC operation pipeline

For HVAC operation with the proposed recognition system, the sampling interval of the vision sensor is 15-min. Each sample consists of a 1-min video, which represents the occupant behaviors indoors for the past 15 minutes. Figure 10 shows a whole sample procedures for such HVAC operations. Different occupant-oriented HVAC operational strategies will be discussed in **energy-simulation** section.

12:00
- video start for 1-min sample

12:01
- video end

12:02-12:14
- the sample video uploaded to the platform
- the program runs for occupant behavior recognition:
  - occupant detection;
  - counting the average as the number of occupants;
  - any other useful video processing (e.g. activity level classification, clothing recognition, etc.)

12:15
- video starts again for next sample
- ....

Figure 10 A sample procedure for HVAC operations with the proposed recognition system

Based on the paper, the development of occupancy pattern recognition system is comprised of occupancy detection, occupancy counting, activity level classification and validation of the recognition system. Sets of 2D image sequence from static cameras were used to interpret the 3D scenes.

The vision-based system was implemented based on Darknet (Redmond & Farhadi, 2016), which is a state-of-art object detection model based on deep learning and the pre-trained model of tiny-yolo-voc based on the training set called VOC. In addition to Darknet, another open source toolbox named OpenCV was also used for activity classification.

- **Preprocessing:**
  Before processing the video, the raw 1-min sample video is preprocessed by converting into grayscale and downsizing into 640x480 pixels.
- **Occupancy detection:**
  Toolbox: Darknet
  Instead of feature selection and detecting objects based on high scorin regions, Darknet designs a single neural network to the whole domain. The network divides the image into regions and predicts bounding boxes and probabilities for each region. The bounding boxes are weighted by the predicted probabilities (Redmond & Farhadi, 2016). Due to the utilization of CUDA, the algorithm can make the use of GPU to speed up the detection while maintain high accuracy. There are different pre-trained models available for different purposes and computing boards. In order to enable the detection in embedded GPU of Jetson TK1, as mentioned before, the smaller weights of neural network of *tiny-yolo-voc* is used with suitable hyperparameters defined in *tiny-yolo-voc.cfg* file. Different from the preset batch size of 8, the batch size has been decreased to 1 so as to fit the embedded platform which has less than 2G memory capacity.
  Besides, in order to feed the video from webcam or input file for object detection in Darknet, two threads are created. One is called *fetch_in_thread* and the other is called *detect_in_thread*.
- **Occupancy counting:**
  Occupancy counting can be realized by count the bounding boxes which are classified as person in each frame. The result of each 1-min sample video is the average number of people detected in each frame over the whole video and is rounded to finite. For instance, if the average number of occupants is detected to be 2.2, the final result of the occupant number for this sample video will be 3. In terms of evaluation method, normalized root mean square error (NRMSE) as well as min/max values as error-bars are both used to evaluate the error/deviation of the number of people detected in each frame.
  Besides, in order to optimize the performances, the method of skipping a number of reasonable frames before occupant detection is used to speed up the detection process.
- **Activity level classification:**
  Toolbox: OpenCV2
  In order to simplify the process of activity level classification, only two classes exist for activity classification, which are if the scene is active or inactive.

Moreover, the activity level classification is integrated into Darknet by creating a new thread called *bg_sub_thread*.

The classification method is based on several assumptions listed below:

- The activity level is positively correlated with the area of foreground consisting of pixels with large deviation and extracted from instantly updated background model;
- The percent of pixels in foreground which are not because of people moving can be omitted;
- Indoor illumination keeps constant for the whole video and background is processed to be smooth and not cluttered;
- The area of a bounding box of a detected occupant can be approximately equal to the area of the occupant body;

Based on these assumptions, background subtraction can be an effective method for generating foreground mask and detecting moving objects in the scene. In OpenCV, a few common subtractors are available, which are k-nearest-neighbors (KNN), mixture of Gaussian (MOG), mixture of Gaussian2 (MOG2) and Godbehere-Matsukawa-Goldberg (GMG). In this project, mixture of Gaussian2 is used for background subtraction task. The pseudo-code for activity level classification is shown below:

1. Converting the current frame into matrix *M*;
2. Applying subtractor of mixture of Gaussian2 to *M*;
3. Calculating the total area of the foreground scene *F*;
4. Calculating the ratio between total foreground area and the total area of bounding boxes in the current frame;
5. If the ratio is larger than 75%(threshold1), the current frame is classified as "active frame";
6. For the whole 1-min sample video, calculating the number of "active frame";
7. Calculating the percent of "active frames" in the current 1-min sample video by getting the ratio between the number of "active frame" and that of the total frames. If the ratio is larger than 75%(threshold2), then this sample video can be classified as active, which represents the activity level for the past 15 minutes.

For activity level classification, the value of threshold1 and threshold2 are both set arbitrarily. More work could be done to get better threshold values. The complete codes for activity level classification can be found in **Appendix A**.

- **Validation of the vision-based occupancy recognition system:**

The classification will be validated with a total of 11 scenarios in different space types, which are lounge, open plan office, conference room and small private office during different time periods. In addition, the proposed occupancy recognition system is also compared with FORK developed by Bosch. The result of activity level classification is evaluated with confusion matrix and the accuracy of occupancy counting with the proposed system is evaluated with average value plus normalized root mean square error.

Meanwhile, the computing board for video processing is Jetson TK1, which is an embedded GPU developed by Nvidia. The configuration of the whole processing unit is shown in Figure **11**.

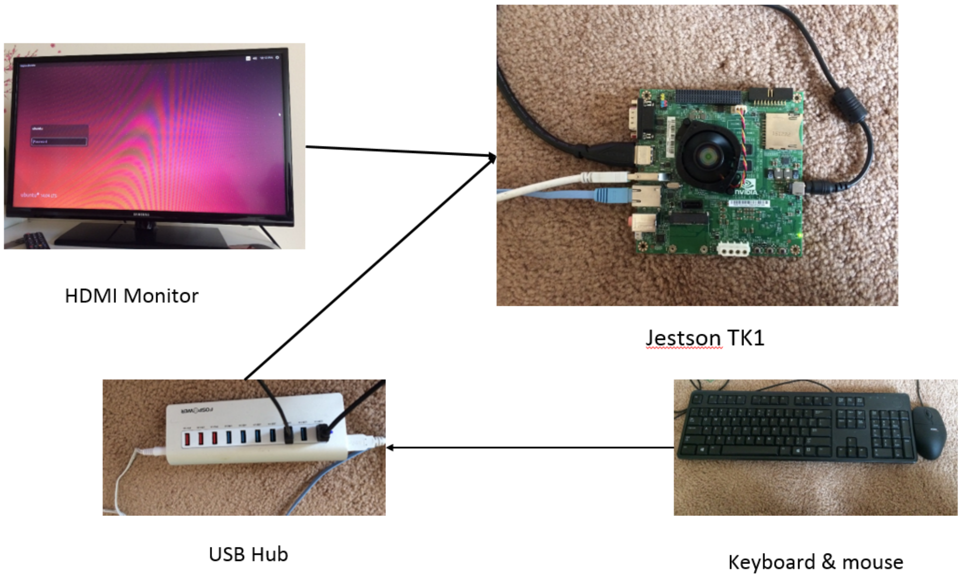


Figure 11 the whole processing unit with embedded platform

As mentioned before, a total of 11 scenarios with the proposed recognition system are tested in different locations. The experimental setups are described as below, respectively.

a) Setup1: Open plan student lounge

- Location: Margret Morrison Carnegie Hall 408(Figure **12**)
- Date: 2/28/2017
- Vision-sensor tool: iPhone 5s
- Mounting position: on the fridge in the south-west corner
- Scenarios:
    - All occupants seated for reading and working;
    - Some of occupants walking around the room and chatting with each other;
    - All occupants having a social party with pizza in the lounge;

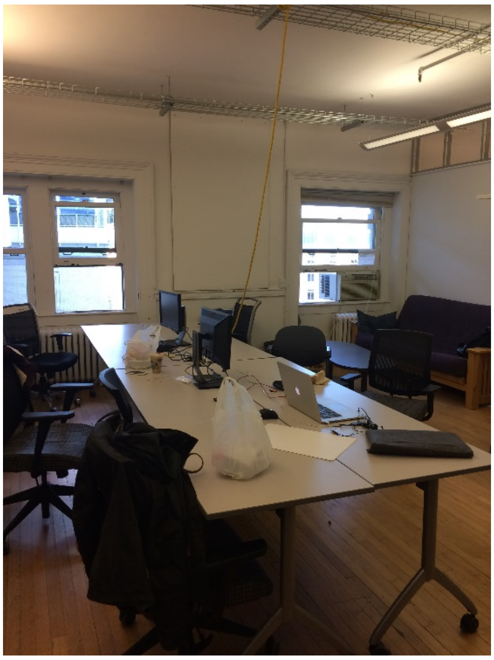


Figure 12 MMCH 408

b) Setup2: Open plan conference room

- Location: Scaife Hall 212(Figure 13)
- Time: 1:30 pm – 2:00 pm, 4/7/2017
- Vision-sensor tool: Nikon
- Mounting position: in the south-west corner of the room
- Scenarios:
  - All occupants seated for reading and working;
  - Some of occupants walking around the room and chatting with each other;
  - All occupants having a social party with pizza in the lounge;

Figure 13 Scaife Hall 212

The second experiment was compared with FORK installed in the room for occupancy estimation. The comparisons between FORK and the proposed system, which refers as "improved Darknet" are listed in Table 3.

Table 3 Comparisons between FORK and the proposed system

	FORK	Proposed system based on improved Darknet
Mounting position	top of the door	corner of the room
Main Algorithm for occupant detection	Multi-level scanning with depth information	Deep learning
Main Activity level classification	Not available	Background subtraction
Sensor	Depth sensor(Kinect)	RGB sensor
Embedded board	Odroid	Jetson TK1
Signal transmit method	Online	Online(currently is offline)
Sampling interval	Continuous	15-min interval

c) Setup3: Small private room

- Location: Hunt library A13(Figure 14)
- Date: 4/22/2017
- Vision-sensor tool: Nikon
- Mounting position: in the corner and 5.7 foot above the floor

Figure 14 Hunt library A13

<u>3) Energy Simulation of HVAC control system with dynamic occupancy profile from Occupancy Simulator:</u>

A research group from University of Southern California (Li et al., 2012) proposed a set of HVAC operational strategies with the availability of occupancy information, which are described as below:

a) maintaining higher temperature in unoccupied areas;
b) maintaining lower ventilation rates in unoccupied areas;
c) supplying airflow and adjusting outdoor airflow rate based on occupancy;
d) responding to dynamic occupant profiles in real time;
e) operating HVAC systems based on occupant preferences;
f) learning energy consumption patterns;
g) increasing the flexibility of control.

In addition to energy-based HVAC control, PMV-based HVAC control has also been investigated to achieve thermal comfort in which metabolic rate, clothing insulation are two factors occupant characteristics inside the building.

Based on the characteristics of vision-based system and HVAC operational strategies mentioned above, an energy simulation of a DOE existing reference building is developed to show the potentials of energy saving and thermal comfort level improvement due to the real-time awareness of occupant information. Instead of being linked with the vision-based system to get dynamic occupant profiles, Occupancy Simulator developed by LBNL was used to generate agent-based occupant profiles and integrate them into the building with two sets of HVAC operational strategies. One is only energy-based HVAC operational strategies and the other is PMV-based HVAC operational strategies with dynamic occupancy schedules.

- Basic building information
    - Location: San Francisco(3D rendering in shown in Figure **15**)

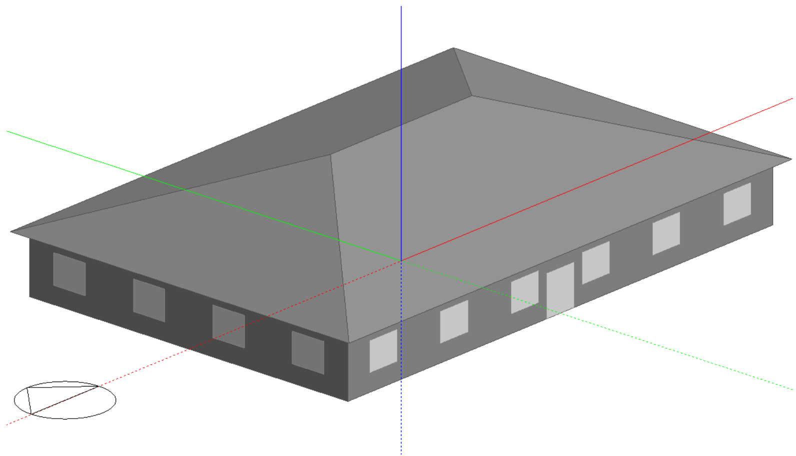


Figure 15 3D rendering of the reference building

    - Building type: Small office building
    - Ground area: 512 m^2
    - Number of floor(floorplan is shown in Figure **16**): 1

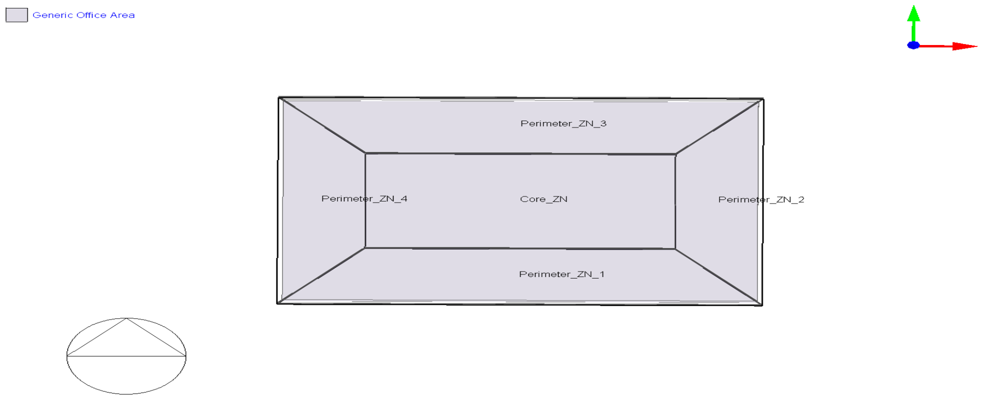


Figure 16 Floorplan of the 1st floor

  - Maximum number of occupants: 28
  - Thermal zones: Attic, Core_ZN, Perimeter_ZN_1, Perimeter_ZN_2, Perimeter_ZN_3 and Perimeter_ZN_4.
  - HVAC system: Packaged DX Coil
    - Cooling: electricity
    - Heating: natural gas

- The baseline occupancy schedule vs the dynamic occupancy schedule
  In Energyplus, occupancy schedules include number of people schedule, activity level schedule and clothing insulation schedule. The method to generate dynamic occupancy schedules are described as below, respectively.
  - The number of people schedule
    The fixed number of people schedule is shown in Figure 17.

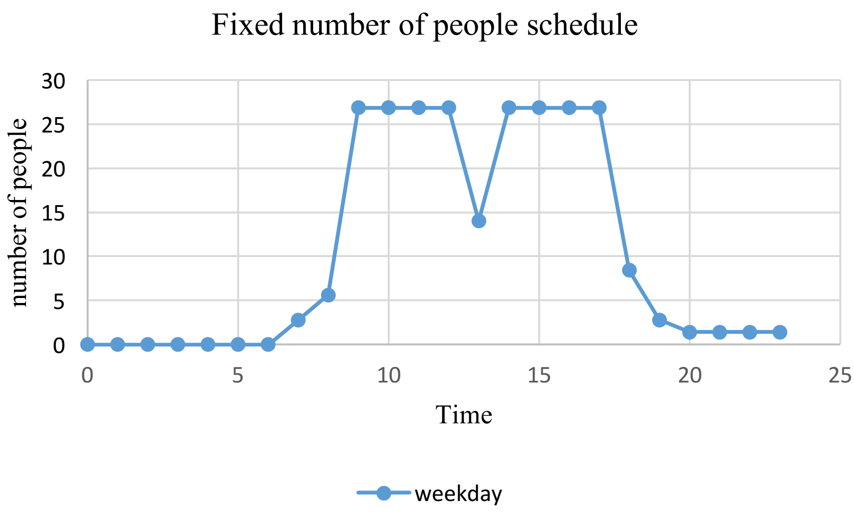


Figure 17 Fixed number of people schedule of baseline

Compared to baseline schedule, the dynamic occupancy schedule created by occupancy simulator is more realistic, thus being more suitable to represent the expected results of the proposed real-time occupant pattern recognition system. The number of occupancy schedule on Mar. 1st, 2010 (Monday) and of a zone is shown in Figure 18.

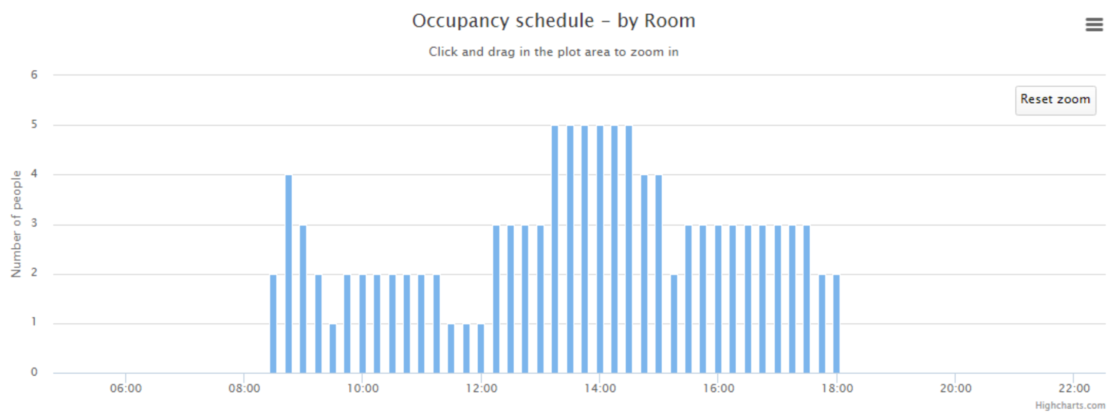


Figure 18 the number of occupant schedule in a zone on Mar, 1st,2010

- Activity level schedule
  The activity level of the baseline building assumes that all the occupants in the building has a constant metabolic rate of 120W/person. However, according to ASHRAE-55, the activity level in an office building changes between 1-1.7 met (1 met = 58.2 W/m2) dynamically. In order to create a dynamic activity level schedule, the metabolic rate value for one timestep is set randomly between 104 W/person and 176.8 W/person based on the assumption human body can be approximated to be 1.8 m^2.
- Clothing insulation schedule
  The clothing insulation schedule of the baseline assumes that only two typical clothing insulation value of 1 clo or 0.5 clo during either winter season or summer season. However, people wear much more diverse clothes than what is assumed in the baseline. Therefore, instead of only using two clothing insulation value, the modified clothing insulation schedule creates a more dynamic clothing insulation schedule, as shown in Table **4**.

Table 4 Dynamic clothing insulation with different seasons

	Clothing insulation [clo]
Spring	0.75
Summer	0.5
Fall	0.75
Winter	1

- Three sets of HVAC operational strategies
  In order to better compare the energy performances and comfort level, it is assumed that the occupancy schedules are all set to be dynamic. However, the differences between different HVAC operational strategies rely on if the strategies respond to the dynamic occupancy information accordingly. Table **5** summarizes three HVAC operational strategies. A total of four schedules may change under different sets of HVAC operational strategies:

a) HVAC operation schedule: the schedule used to control if HVAC system should be on or off;
b) Minimum outdoor air rate schedule: the schedule used to control the portion of the outdoor air rate in supply air;
c) Heating setpoint schedule: the schedule used to determine minimum indoor air temperature without heating. In other words, if the indoor air temperature is lower than heating setpoint, then heating system should be on;
d) Cooling setpoint schedule: the schedule used to determine maximum indoor air temperature without cooling. In other words, if the indoor air temperature is higher than cooling setpoint, then cooling system should be on.

Table 5 Comparisons between three HVAC operational strategies

	Baseline	**Energy-based HVAC control**	**PMV-based HVAC control**
HVAC Operation Schedule	Always on during fixed periods	if occupied, on; else, off	if occupied, on; else, off
Minimum Outdoor Air Rate Schedule	Fixed schedule	change proportional to occupant density	change proportional to occupant density
Heating Setpoint Schedule	Fixed schedule	Fixed schedule	PMV-based control algorithm
Cooling Setpoint Schedule	Fixed schedule	Fixed schedule	PMV-based control algorithm

PMV-based HVAC control algorithm is based on ASHRAE 55 and CBE Thermal Comfort Tool (Hoyt et al.) where it recommends the acceptable range of PMV is between -0.5 and 0.5. PMV-based HVAC control is mainly to create dynamic heating and cooling setpoint schedules based on calculated PMV. The proposed PMV-based HVAC control is based on several assumptions as follow:

- The indoor relative humidity is constant to be 60% for all the occupied zones;
- The indoor air velocity is constant to be 0.2 m/s;
- The mean radiant air temperature doesn't differ from indoor air temperature; thus indoor air temperature can be approximated to be mean radiant air temperature;
- The external work is 0;
- As mentioned before, activity level in office room is between 1-1.7 met;
- As mentioned before, clothing insulation for spring (Apr.-Jun.) or fall (Oct.-Dec.) is 0.75 clo while winter (Jan.-Mar.) or summer (Jul.-Aug.) is 1 or 0.5 clo, respectively.

As shown in **Figure 19**, the psychometric chart visually shows generally how heating and cooling setpoints are set based on PMV calculation.

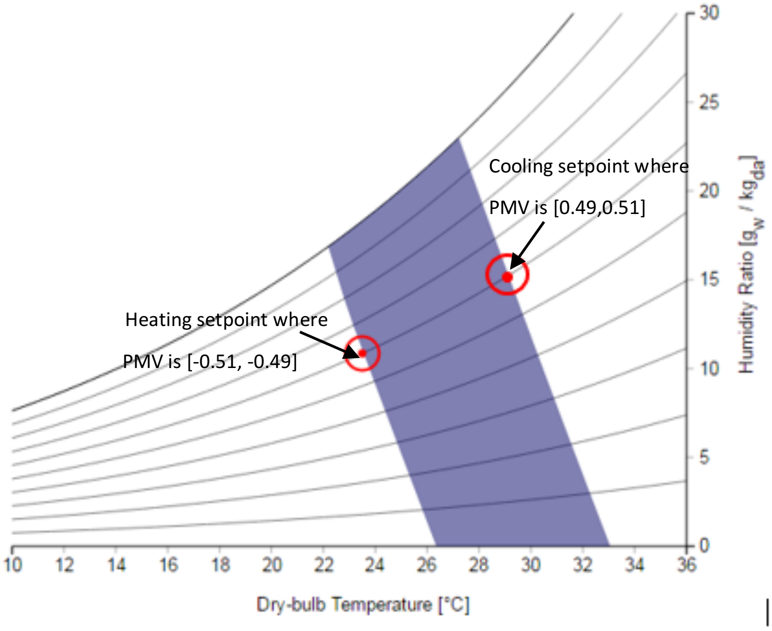


Figure 19 Psychometric chart with comfort range

The pseudo-code for PMV-based HVAC control is shown as below:

1. Initialization:
   Arbitrarily set the initial heating or cooling setpoint as a value *temp0*, which is larger than 0;
2. Calculating PMV beginning with *temp0* and do the recursion to get heating and cooling setpoints:
   a. For heating setpoint, if calculated PMV is between -0.51 and -0.49, then set the heating setpoint to be *temp0*;
   b. For heating setpoint, if calculated PMV is larger than -0.49, then decrease *temp0* by 0.1 ℃ and do the recursion;
   c. For heating setpoint, if calculated PMV is smaller than -0.51, then increment *temp0* by 0.1 ℃ and do the recursion;
   d. For heating setpoint, if the calculated PMV of *temp0-0.1* is smaller than -0.51 while the calculated PMV of *temp0+0.1* is larger than -0.49, then set the heating setpoint to be *temp0*. This is to avoid the recursion to be infinite due to the endless discrete jump between *temp0+0.1* and *temp0-0.1*;
   e. If the calculated heating setpoint is below the default setback, which is 15.6 C, the heating setpoint is set to be 15.6 ℃;
   f. For cooling setpoint, the procedure is almost the same except that the calculated PMV of the cooling setpoint should be within the range between 0.49 and 0.51 and if the calculated cooling setpoint is higher than the default setback, which is 26 ℃, the cooling setpoint is set to be 26 ℃;

The complete codes of PMV-based HVAC control can be found in **Appendix B**.

Finally, with the dynamic occupancy schedules consisting of the number of people schedule, activity level schedule, clothing insulation schedule as well as three sets of HVAC operational strategies, the energy simulation software of Energyplus can be used to simulate the annual energy consumption and occupant comfort level and the potential benefits from dynamic occupancy profiles which can be achieved from proposed real-time occupant pattern recognition system can be evaluated.

4) Result analysis and project report completion:

In order to validate the reliability of the proposed vision-based system for occupant pattern recognition and see if the integrated control system has better performances than the existing HVAC control system from perspectives of both energy reduction and occupant comfort, a set of comparisons were conducted.

Comparison with the actual occupant patterns:

- the recognized activity level vs the actual activity level of occupants with confusion matrix;
- the number of detected occupant vs the actual number of occupant with min/max error for all scenarios;
- the comparison between FORK and the proposed recognition system in terms of accuracy of occupancy counting;
- the time for video processing vs the accuracy for optimization by skipping frames for the best scenario;

Comparison among three sets of HVAC operational strategies:

- The PMV-based HVAC heating and cooling setpoint control on a selected day vs the occupant density;
- The validation of the proposed PMV-based HVAC control by showing the calculated PMV over time with Energyplus;
- Comparison of comfortable days throughout a year among these three sets (assumed that the number of occupied hours on weekday is 8 hours);
- Comparison of the simulated annual energy consumption per conditioned area among these three sets;
- Comparison of whole building facility total HVAC electric demand power during August, 2010 based on the three different sets of HVAC operational strategies;

Last but not least, a complete report was submitted to fulfill the requirement of master program in building performance and diagnostics.

4 Result analysis

- **Development of real-time vision-based occupant recognition system**
  a) Experimental setup 1

  Table 6 shows the accuracy of occupancy counting in the experiment conducted in the open plan area of lounge. It can be observed that the scenario of people walking and chatting has the best performance with 100% accuracy.

Table 6 The performance of the proposed recognition system in experimental setup 1

Number	frame rate[fps]	activity classification	Activity ground truth	Average Occupant Number	Ground truth	NRMSE	Accuracy
1	[6,7]	inactive	inactive	4	7	64%	57%
2	[6,7]	active	active	7	7	18%	100%
3	[6,7]	active	active	8	10	30%	80%

b) Experimental setup 2

Table 7 shows the performance of the proposed recognition system and comparisons with FORK, the occupancy estimation tool developed by Bosch. As shown in the table, scenario 5 has the highest accuracy of occupancy counting with least NRMSE of 27.7%. However, compared with FORK, the average accuracy among these scenarios is not so high as that of FORK. Despite this, the proposed recognition system can realize activity level classification since the sensor is installed inside the room while FORK hasn't realized the function since it is mounted at the door and only estimates the number of occupants entering or leaving the room. Moreover, the frame rate of the proposed system is twice as FORK which may because of the camera quality.

Table 7 The performance in experimental setup 2 and comparisons with FORK

umber	Frame rate		Activity classification			Average Occupant Number		Ground truth		NRMSE		Accuracy	
	FORK	Darknet	FORK	Darknet	Ground truth	FORK	Darknet	FORK	Darknet	FORK	Darknet	FORK	Darknet
1	[4,9]	[12,15]	None	inactive	inactive	8	7	8	9	None	62.34%	100%	77.78%
2	[4,9]	[12,15]	None	inactive	inactive	8	7	8	9	None	43.51%	100%	77.78%
3	[4,9]	[12,15]	None	inactive	inactive	7	6	8	9	None	59.41%	87.5%	66.67%
4	[4,9]	[12,15]	None	inactive	inactive	8	8	8	9	None	27.92%	100%	88.89%
5	[4,9]	[12,15]	None	inactive	inactive	8	9	8	9	None	27.7%	100%	100%
6	[4,9]	[12,15]	None	inactive	inactive	8	7	8	9	None	59.43%	100%	77.78%

c) Experimental setup 3

Table 8 shows the performance of the proposed system in setup 3. As shown in the table, scenario 1 has lower accuracy than scenario 2. The average number of occupants is 2.11 where the rounded result is 3. Since in private room, there are usually only a few people inside, thus one increment will affect a lot on accuracy. Therefore, the occupant presence matters more than number of occupants as it determines if HVAC system should be operated and the ventilation for small room consumes less energy than open plan areas where many people are inside. Therefore, for small private room, as long as occupant presence can be detected, it is acceptable if the accuracy of occupancy counting is not great.

Table 8 The performance of the proposed recognition system in experimental setup 3

Number	frame rate[fps]	activity classification	Activity ground truth	Average Occupant Number	Ground truth	NRMSE	Accuracy
1	[12,15]	inactive	inactive	3	2	23.28%	57%
2	[12,15]	inactive	active	2	2	27.75%	100%

Figure 20 Two students seated at the table:

In addition to occupancy counting, Table 9 shows performance of activity level classification. As shown in the table, the sum of true-positive predictions (upper-left cell) and true-negative (lower-right cell) predictions be over 90% of the total predictions. It indicates that the activity level classification can predict activity level effectively.

Table 9 Confusion matrix of activity level classification

	predicted class: active	predicted class: inactive
actual class: active	2	0
actual class: inactive	1	8

The following figure shows comparisons between the number of detected occupant and the actual number of occupant with error bars for a total of 11 scenarios. As shown in the figure, the difference between the max/min values and the average value vary a lot among different scenarios. Unlike evaluation result with NRMSE, the scenario with better accuracy may not have smaller error ranges, which indicates the detection has several outliers and the performance of occupant detection of each frame is not stable. Therefore, the algorithm of the proposed recognition system needs to be improved in the future work.

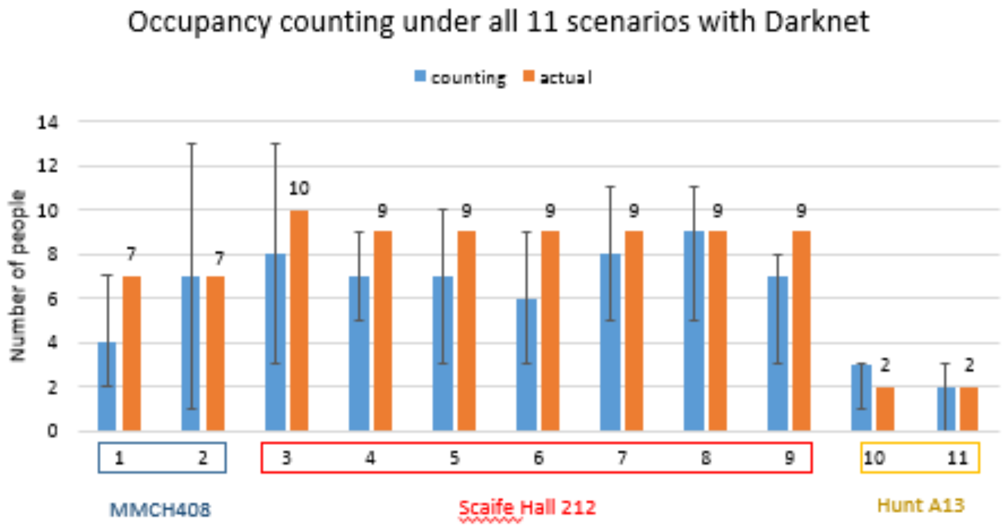


Figure 21 occupancy counting of all 11 scenarios

Figure 22 shows the box plot of the performance of detected occupant number with the proposed recognition system in Scaife Hall. As mentioned before, scenario 5 has the best performance and the corresponding scene is shown in Figure 23. Since there are not many occlusions between each person, the proposed system can detect each occupant with higher prediction probability.

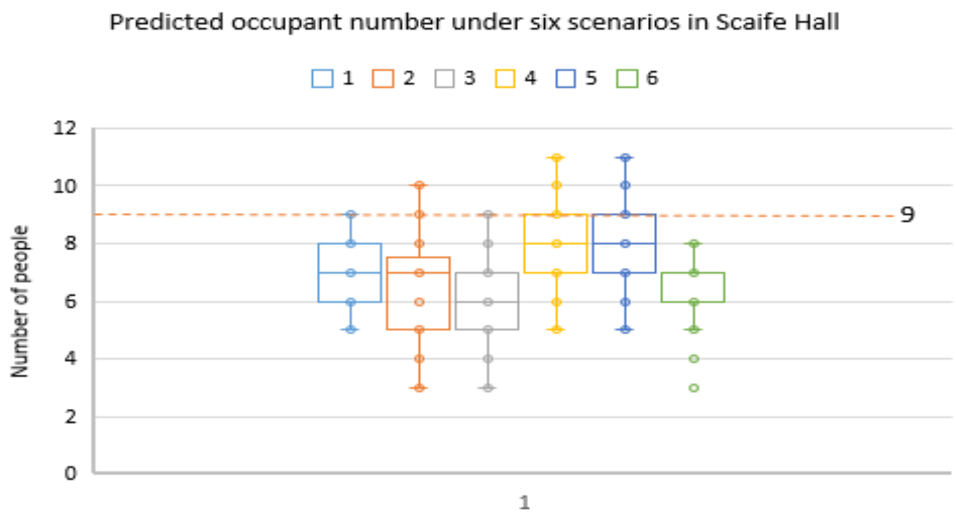


Figure 22 Predicted occupant number under six scenarios in Scaife Hall

Figure 23 The scene of the best scenario without many occlusions

Besides accuracy, speed is also an important metric to optimize. Figure 24 shows the comparisons among three different cases where the method of skipping frame is used. As shown in the figure, skipping a number of frames doesn't compromise the performance of occupancy counting for the best scenario. It indicates that it has the potential to speed up the process if the baseline is able to get a good accuracy.

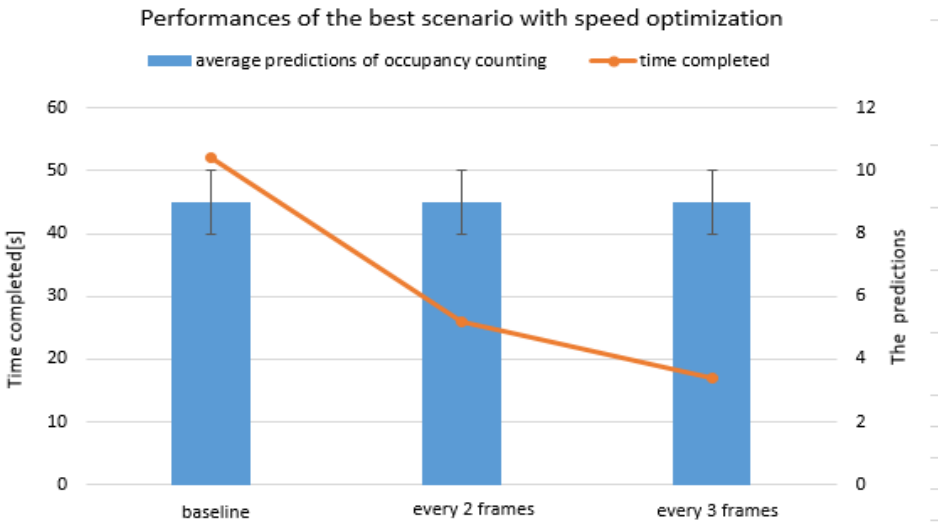


Figure 24 performances of the best scenarios with speed optimization

Lastly, Figure 25 shows the comparison of occupancy counting between FORK and the proposed system. As mentioned before, for most cases, the accuracy with FORK is higher than that with the improved Darknet. The proposed recognition system has an advantage of activity level classification over FORK, which is important for indoor thermal comfort control.

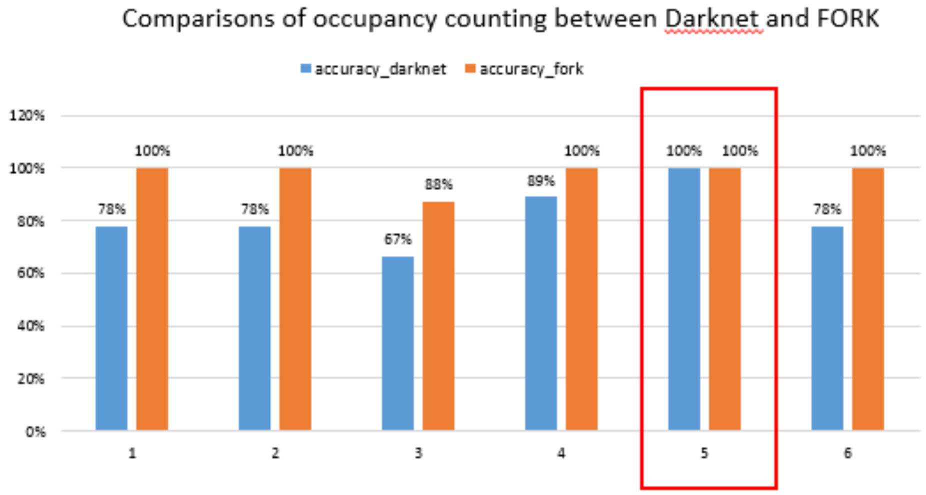


Figure 25 Comparisons of occupancy counting between Darknet and FORK

- **Comparison among three sets of HVAC operational strategies:**

As mentioned before, PMV-based HVAC operational strategies set heating and cooling setpoints according to PMV calculation with the inputs of dynamic occupancy schedules of activity level and clothing insulation. Figure 26 shows the changes of heating and cooling setpoints on August, 7th, 2010 with PMV-based HVAC operational strategies. As shown in the figure, during unoccupied time, heating and cooling setpoints are both set constantly as setback temperature while during occupied time, heating and cooling setpoints are more dynamic. However, since calculation of PMV doesn't take occupant density into account and in Energyplus, the smallest unit is zone. Therefore, each person in one zone is assumed to have the same thermal sensation in PMV-based HVAC control. The proposed PMV-based HVAC operational strategy is validated with the PMV calculation by Energyplus with the inputs of air temperature, mean radiant temperature, relative humidity, air speed, metabolic rate and clothing insulation based on each timestep. The result in Figure 27 shows the calculated PMV with Energyplus on the same day. In order to make the figure clearer, the PMV value during unoccupied time is set as 0. According to the figure, it can be observed that during occupied time, PMV can be controlled within the range of [-1,1]. In other words, people in the building feel thermally comfortable during the working hours with the proposed PMV-based HVAC heating and cooling setpoint control.

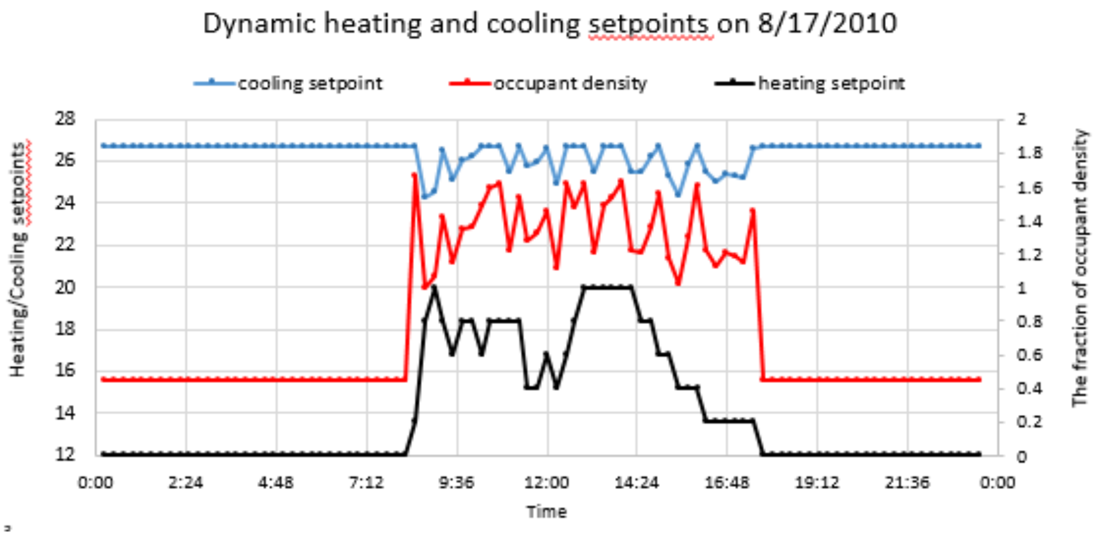


Figure 26 Dynamic heating and cooling setpoints

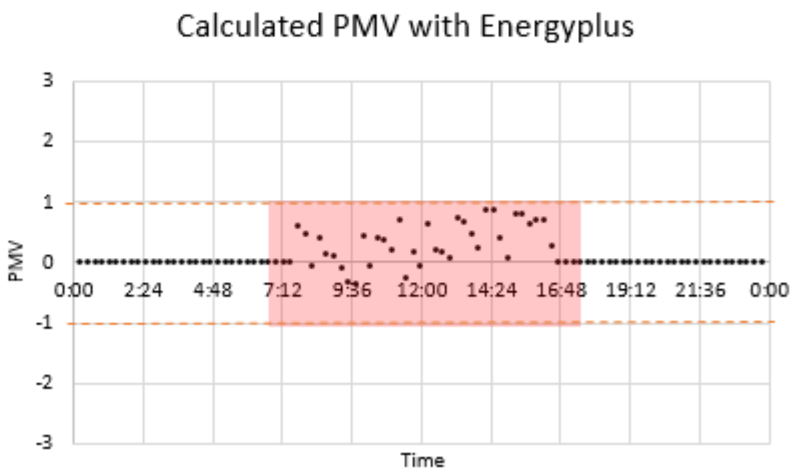


Figure 27 Calculated PMV with Energyplus

In order to compare performances of three sets of HVAC operational strategies from perspective of thermal comfort level and energy consumption, Figure 28 and Figure 29 show comparisons of comfortable days and energy consumption per conditioned area throughout a year. It can be observed from the figures that PMV-based heating and cooling setpoints control strategy (referring as dynamic setpoint in the figure) can achieve the largest amount of comfortable days while consume least amount of energy annually. Compared with baseline, PMV-based HVAC operational strategy has 6% more comfortable days and saves 10% of energy consumption.

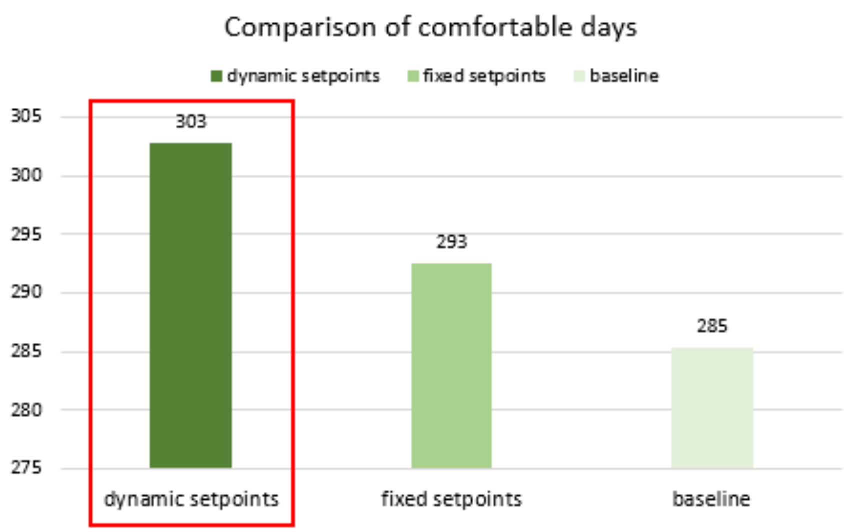


Figure 28 comparisons of comfortable days among three sets of HVAC operational strategies(assumed that the number of occupied hours on weekday is 8 hours)

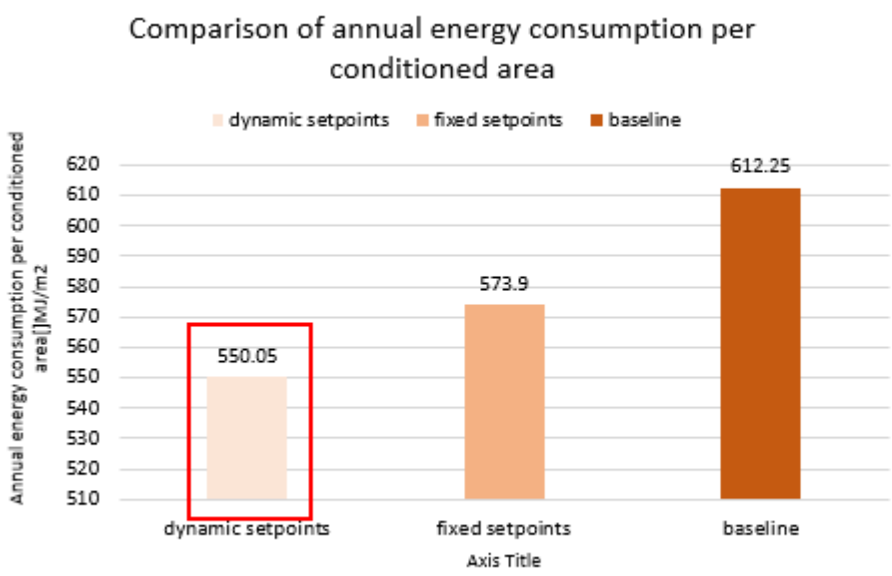


Figure 29 Comparisons of annual energy consumption per conditioned area with three sets of HVAC operational strategies

Last but not least, Figure 30 shows the comparison of whole building facility total HVAC electric demand power during August, 2010 between baseline HVAC operational strategy and PMV-based HVAC operational strategy. Since each zone is equipped with a single DX-coil package where cooling is supplied with electricity, the figure indicates the energy end use during summer. It can be observed that during peak period, the building with baseline strategy has higher HVAC electric demand than with dynamic-setpoint strategy. Therefore, HVAC operation responsive to occupant profiles in real time has a great potential to save energy and improve occupant comfort level.

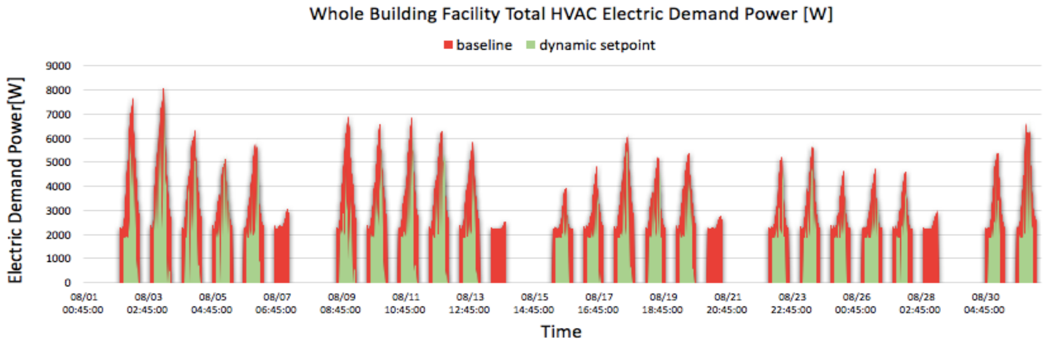


Figure 31 Whole Building Facility Total HVAC Electric demand power

Figure 30 Comparison of whole building facility total HVAC electric demand power between baseline and HVAC operational strategies with dynamic setpoint control

5 Conclusions:

This thesis contributes to the development of a new occupancy pattern recognition system with the improved Darknet which realizes occupancy counting and activity level classification so that the dynamic occupancy schedules can be achieved in real-time with high accuracy to enable occupant-oriented HVAC operational strategies. Several conclusions can be drawn based on findings above.

- The proposed real-time occupancy pattern recognition system of the improved Darknet has accuracy over 90% without many occlusions in terms of occupancy counitng; However, the average accuracy is less than that of FORK;
- The proposed real-time occupancy pattern recognition system of the improved Darknet has the ability to realize activity level classification where the accuracy reaches over 90%.
- The energy simulation of a DOE reference building illustrates PMV-based HVAC operational strategy which calculates PMV to determine heating and cooling setpoints can achieve more comfortable days and consume least amount of energy compared with HVAC operational strategy with fixed heating/cooling setpoint schedules while occupant density and occupant presence is considered as well as the baseline HVAC operational strategy.
- It is important to use responsive HVAC operational strategies to real-time occupant related information so as to improve building performances, including energy savings and occupant comfort level.

6 Discussions:

- Development of real-time vision-based occupant recognition system
  - Sensor configuration

    In terms of occupancy sensing technology, three key technical factors have to be taken into consideration: accuracy, privacy and intrusiveness

Accuracy is highly dependent on the effectiveness of the algorithm for recognition. Currently, Darknet has been a powerful analysis tool for detection. However, since the open library detects a total of 20 classes, thus the neural network being complex and the weights being large enough to ensure acceptable accuracy. However, if the neural network is only for occupant detection, much more specific neural network architecture can be designed. In addition, since the activity level classification is based on the bounding boxes of detected occupants, it is also necessary to ensure the size of the bounding boxes is close to the size of human bodies.

Privacy is another key issue for vision-based occupant recognition system. However, based on the current survey, approximately 48% of office spaces are installed with surveillance cameras. Nevertheless, since the purpose of surveillance cameras differ from that of vision sensor and the position of the vision sensor matters a lot, it is also necessary to see the potential connections between the surveillance cameras and the vision sensors.

Last but least is the intrusiveness. Luckily, compared with wearable devices such as bio-sensors or RFID, since camera is installed in the ambient environment, it is not that intrusive. However, compared with environmental sensor fusion, it is still a bit more intrusive.

Besides these three key technical issues, the vision sensor could have further ability to detect other characteristics of the occupants such as gender and age. However, since such features may need closer observation such as facial recognition, which might compromise the privacy, the further evaluation between the cost and the benefit are necessary. Meanwhile, in terms of activity level classification, the current of background subtraction has many assumptions. Therefore, it may be better to develop action recognition system to recognize the activity level in a more accurate way.

- Sensor network

Once the vision-sensor is developed, in order to implement it into the real building for HVAC operation, it is necessary to understand the corresponding sensor network at least in terms of topology, network architecture and power management.

Sensor topology for such vision sensor may not be so complex since each building has its own sensor network and the only communication occurs between the neighboring sensor nodes and between the sensor and the controller of the HVAC terminal like VAV box.

However, like many sensor network, architecture could be complex. For instance, whether sensors need to communicate in a synchronous way or an asynchronous way and how to encode the message and how large the data storage should be.

Last but not least is the power management. Currently, the proposed recognition system is processing videos offline with the embedded board plugged through electric outlet. However, in order to implement the sensor, it is necessary to design a processing unit which can be powered onboard and consumes less energy than other sensors.

- Scalability

  Currently, the proposed recognition system is designed for office building, particularly the open plan areas where occupant density matters for ventilation rates. However, it could be also meaningful to consider other building types such as sports center or hospital. A good sensing system is able to be implemented in most of building types and provide real-time occupant information for dynamic HVAC operations.

- Occupant-oriented HVAC control:
  - PMV-based HVAC control

    The proposed PMV-based HVAC control has a lot of assumptions as well, particularly the relative humidity keeps constant. Even if in office buildings, it is acceptable to keep the humidity constant, it is still better to get the real-time humidity data. In addition, according to ASHRAE 55, the measurement of the clothing and activity level should be the mean value over a period of 0.5 to 1.0 hour immediately prior to measuring the thermal parameter. Therefore, the current 1-minute-long video for recognition may not be enough for clothing and activity level measurement. Last but not least, since Energyplus set zone as its smallest unit, PMV-based setpoint control cannot reflect the different thermal sensation of individuals in one zone.

- Cost

  Cost analysis for a new system is necessary to evaluate if it is worth developing such recognition system for long-term operation. Due to complexity of sensor configuration, network and control, it may result in high initial investment. However, since occupant-oriented HVAC operation indeed can bring about a certain amount of energy savings and increase comfort level, it is still worthwhile to make effort to developing such smart recognition system.

7 Future work:

Many future work shall be done to improve real-time vision-based occupancy pattern recognition system and make the use of such real-time occupancy information. As discussed in the previous section, several key points shall be completed:

- Sensor configuration to improve accuracy of occupancy counting and activity level classification as well as mitigate the issue of privacy;
- More patterns could be realized such as clothing;
- Design of sensor network for HVAC system;
- Better control logic using real-time occupant information such as predicted model control;
- Cost analysis of the proposed recognition system;
- Implementation of such occupancy sensor network in different building types

8 References:

Antonakaki, P., Kosmopoulos, D., & Perantonis, S. J. (2009). Detecting abnormal human behaviour using multiple cameras. *Signal Processing*, *89*(9), 1723–1738. https://doi.org/10.1016/j.sigpro.2009.03.016

Benezeth, Y., Laurent, H., Emile, B., & Rosenberger, C. (2011). Towards a sensor for detecting human presence and characterizing activity. *Energy and Buildings*, *43*(2–3), 305–314. https://doi.org/10.1016/j.enbuild.2010.09.014

Cao, X. (2012). Face Alignment by Explicit Shape Regression, 2887–2894.

Chen, Y., Luo, X., & Hong, T. (2016). An Agent-Based Occupancy Simulator for Building Performance Simulation. *ASHRAE Annual Conference*.

Choi, J. H. (2010). CoBi: bio-sensing building mechanical system controls for sustainably enhancing individual thermal comfort.

Dong, B. (2010). Integrated Building Heating, Cooling and Ventilation Control. *Thesis*, 1–174. Retrieved from http://repository.cmu.edu/dissertations/4/

Efros, A., Berg, A., Mori, G., & Malik, J. (2003). Recognizing action at a distance. *Proceedings Ninth IEEE International Conference on Computer Vision*, (October), 726–733. https://doi.org/10.1109/ICCV.2003.1238420

Hoai, M., Lan, Z.-Z., & Torre, F. D. la. (2011). Joint segmentation and classification of human actions in video. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 3265–3272. https://doi.org/10.1109/CVPR.2011.5995470

Hoyt Tyler, Schiavon Stefano, Piccioli Alberto, Moon Dustin, and Steinfeld Kyle, 2013, CBE Thermal Comfort Tool. Center for the Built Environment, University of California Berkeley, http://cbe.berkeley.edu/comforttool/

Kazemi, V., & Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1867–1874. https://doi.org/10.1109/CVPR.2014.241

Li, N., Calis, G., & Becerik-Gerber, B. (2012). Measuring and monitoring occupancy with an RFID based system for demand-driven HVAC operations. *Automation in Construction*, *24*, 89–99. https://doi.org/10.1016/j.autcon.2012.02.013

Malmaud, J., Huang, J., Rathod, V., Johnston, N., Rabinovich, A., & Murphy, K. (2015). What's Cookin'? Interpreting Cooking Videos using Text, Speech and Vision. *Annual Conference of the North American Chapter of the ACL*, 143–152. Retrieved from http://arxiv.org/abs/1503.01558%5Cnhttp://arxiv.org/abs/1503.0155%5Cnhttp://arxiv.org/abs/1503.1558

Moeslund, T. B., Hilton, A., & Kr??ger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, *104*(2–3 SPEC. ISS.), 90–126. https://doi.org/10.1016/j.cviu.2006.08.002

Munir, S., Arora, R., et al. "Real-Time Fine Grained Occupancy Estimation using Depth Sensors on ARM Embedded Platforms." In RTAS, Pittsburgh, USA, 2017

Redmon, J., & Farhadi, A. (2016). YOLO9000: Better, Faster, Stronger. arXiv preprint arXiv:1612.08242.

Standard, A. S. H. R. A. E. (2010). Standard 55-2010:"Thermal Environmental Conditions for Human Occupancy"; ASHRAE. Atlanta USA.

Shih, H. C. (2014). A robust occupancy detection and tracking algorithm for the automatic monitoring and commissioning of a building. *Energy and Buildings*, *77*(May 2013), 270–280. https://doi.org/10.1016/j.enbuild.2014.03.069

Shih, O., & Rowe, A. (2015, April). Occupancy estimation using ultrasonic chirps. In Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems (pp. 149-158). ACM.

Sun, K., & Hong, T. (2017). A simulation approach to estimate energy savings potential of occupant behavior measures. Energy and Buildings, 136, 43-62.

Xiong, X., & De La Torre, F. (2013). Supervised descent method and its applications to face alignment. *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 532–539. https://doi.org/10.1109/CVPR.2013.75

Zhao, J. (2015). Design-Build-Operate Energy Information Modeling for Occupant-Oriented Predictive Building Control. Retrieved from http://repository.cmu.edu/cgi/viewcontent.cgi?article=1472&context=dissertations

The latest on workplace monitoring and surveillance, available at: http://www.amanet.org/training/articles/The-Latest-on-Workplace-Monitoring-and-Surveillance.aspx

The inside story: A guideline to indoor air quality, EPA, 2017. available at: https://www.epa.gov/indoor-air-quality-iaq/inside-story-guide-indoor-air-quality

9 Appendix:

Appendix A: Activity-level classification
(it is integrated into demo.c from Darknet)

```
#include "network.h"
#include "detection_layer.h"
#include "region_layer.h"
#include "cost_layer.h"
#include "utils.h"
#include "parser.h"
#include "box.h"
#include "image.h"
#include "demo.h"
#include <sys/time.h>
#include <unistd.h>


#define FRAMES 3


#ifdef OPENCV
#include "opencv2/highgui/highgui_c.h"
#include "opencv2/imgproc/imgproc_c.h"
#include "opencv2/opencv.hpp"
```

```
//#include "opencv2/imgproc/imgproc.hpp"
#include "opencv2/video/video.hpp"
//C
#include <stdio.h>
//C++
#include <iostream>
#include <sstream>


using namespace cv;
using namespace std;
//image get_image_from_stream(CvCapture *cap);
image get_image_from_stream(IplImage *src);


static char **demo_names;
static image **demo_alphabet;
static int demo_classes;


static float **probs;
static box *boxes;
static network net;
static image in   ;
static IplImage *src;
static image in_s ;
static image det  ;
static image det_s;
static image disp = {0};
static CvCapture * cap;
//static VideoCapture cap1;
static float fps = 0;
static float demo_thresh = 0;
static float demo_hier_thresh = .5;


static float *predictions[FRAMES];
static int demo_index = 0;
static image images[FRAMES];
static float *avg;
//an int array to store occupant number in the image of a video
static int *countpv;
static int countp;
// the total areas of boxes for detected persons
```

```
static int total_area;
//write to the file
static FILE *output;
//count the total frame
static int count_f;
// count the active frame
static int is_active=0;
//slow down the frame rate when necessary
static double target_frame_rate;
static double target_frame_time;


// Global variables for background subtraction
Mat frame1; //current frame
Mat fgMaskMOG2; //fg mask fg mask generated by MOG2 method
Ptr<BackgroundSubtractor> pMOG2; //MOG2 Background subtractor
char keyboard; //input from keyboard
int history = 1;
float varThreshold = 16;
bool bShadowDetection = false;


void *fetch_in_thread(void *ptr)
{
    src = cvQueryFrame(cap);
    frame1 = cvarrToMat(src);
    in = get_image_from_stream(src);
    //hardcode due to limited memory of Jetson TK1!
    if(count_f > 700){
       if(is_active > count_f * .75){
           fprintf(output,"This sample is classified as
active!!");
           fclose(output);
       }
       else{
           fprintf(output,"This sample is classified as
nonactive!!");
           fclose(output);
       }
    }


    if(!in.data){
        fprintf(output,"The number of total frames:%d\n",count_f);
```

```
        //activity level classification, threshold2
        if(is_active > count_f* .75)
            fprintf(output,"This sample is classified as
acitve!!\n");
        else
            fprintf(output,"This sample is classified as
nonactive!!\n");
        fclose(output);
        error("Stream closed.");
    }
    in_s = resize_image(in, net.w, net.h);
    return 0;
}


void *detect_in_thread(void *ptr)
{
    float nms = .4;


    layer l = net.layers[net.n-1];
    float *X = det_s.data;
    float *prediction = network_predict(net, X);


    memcpy(predictions[demo_index], prediction,
l.outputs*sizeof(float));
    mean_arrays(predictions, FRAMES, l.outputs, avg);
    l.output = avg;


    free_image(det_s);
    if(l.type == DETECTION){
        get_detection_boxes(l, 1, 1, demo_thresh, probs, boxes,
0);
    } else if (l.type == REGION){
        get_region_boxes(l, 1, 1, demo_thresh, probs, boxes, 0, 0,
demo_hier_thresh);
    } else {
        error("Last layer must produce detections\n");
    }
    if (nms > 0) do_nms(boxes, probs, l.w*l.h*l.n, l.classes,
nms);
    printf("\033[2J");
```

```
    printf("\033[1;1H");
    printf("\nFPS:%.1f\n",fps);
    printf("Objects:\n\n");


    images[demo_index] = det;
    det = images[(demo_index + FRAMES/2 + 1)%FRAMES];
    demo_index = (demo_index + 1)%FRAMES;


    countp=0;
    total_area=0;
    draw_detections(det, l.w*l.h*l.n, demo_thresh, boxes, probs,
demo_names, demo_alphabet, demo_classes,&countp,&total_area);
    return 0;
}
// add background subtraction as a thread & change demo.h
void *bg_sub_in_thread(void *ptr){
        pMOG2->operator()(frame1,fgMaskMOG2);
        int total_pixel = fgMaskMOG2.rows * fgMaskMOG2.cols;
        // percent of foreground with detected people based on
areas of bounding boxes of all detected persons
        double percent = (double) countNonZero(fgMaskMOG2) /
(double) total_area;
        printf("percent of foreground: %f\n",percent);
        // threshold1
        if(percent > 0.75)
            is_active ++;
    return 0;
}


double get_wall_time()
{
    struct timeval time;
    if (gettimeofday(&time,NULL)){
        return 0;
    }
    return (double)time.tv_sec + (double)time.tv_usec * .000001;
}
```

```
void demo(char *cfgfile, char *weightfile, float thresh, int
cam_index, const char *filename, char **names, int classes, int
frame_skip, char *prefix, float hier_thresh)
{

    image **alphabet = load_alphabet();
    int delay = frame_skip;
    demo_names = names;
    demo_alphabet = alphabet;
    demo_classes = classes;
    demo_thresh = thresh;
    demo_hier_thresh = hier_thresh;
    printf("Demo\n");
    net = parse_network_cfg(cfgfile);
    //frame number
    int frame;
    if(weightfile){
        load_weights(&net, weightfile);
    }
    set_batch_network(&net, 1);


    srand(2222222);


    if(filename){
        printf("video file: %s\n", filename);
        cap = cvCaptureFromFile(filename);
        frame =
(int)cvGetCaptureProperty(cap,CV_CAP_PROP_FRAME_COUNT);
       printf("total frame:%d\n",frame);
        //allocate the array by doubling the frame number
        countpv = (int*)calloc(2*frame,sizeof(int));
    }else{
        cap = cvCaptureFromCAM(cam_index);
    }


    if(!cap) error("Couldn't connect to webcam.\n");
    target_frame_rate = cvGetCaptureProperty(cap,CV_CAP_PROP_FPS);
    target_frame_time = 1.0/target_frame_rate;
    printf("target frame rate:%f\n",target_frame_rate);
```

```
    layer l = net.layers[net.n-1];
    int j;


    avg = (float *) calloc(l.outputs, sizeof(float));
    for(j = 0; j < FRAMES; ++j) predictions[j] = (float *)
calloc(l.outputs, sizeof(float));
    for(j = 0; j < FRAMES; ++j) images[j] = make_image(1,1,3);


    boxes = (box *)calloc(l.w*l.h*l.n, sizeof(box));
    probs = (float **)calloc(l.w*l.h*l.n, sizeof(float *));
    for(j = 0; j < l.w*l.h*l.n; ++j) probs[j] = (float
*)calloc(l.classes, sizeof(float));


    pthread_t fetch_thread;
    pthread_t detect_thread;
    pthread_t bg_sub_thread;


    fetch_in_thread(0);
    det = in;
    det_s = in_s;


    fetch_in_thread(0);
    detect_in_thread(0);
    disp = det;
    det = in;
    det_s = in_s;


    for(j = 0; j < FRAMES/2; ++j){
        fetch_in_thread(0);
        detect_in_thread(0);
        disp = det;
        det = in;
        det_s = in_s;
    }


    int count = 0;
    count_f = 0;
```

```
        if(!prefix){
            cvNamedWindow("Demo", CV_WINDOW_NORMAL);
            cvMoveWindow("Demo", 0, 0);
            cvResizeWindow("Demo", 1352, 1013);
        }


        double before = get_wall_time();
        output = fopen("count.txt","w+");
        //background subtractor
        pMOG2 = new
    BackgroundSubtractorMOG2(history,varThreshold,bShadowDetection);
    //MOG2 approach


        while(1){
            ++count;
            ++count_f;
            printf("frame ID:%d\n",count);
            if(1){
                    if(pthread_create(&fetch_thread, 0,
    fetch_in_thread, 0)) error("Thread creation failed");
                if(count%2==0){
                    if(pthread_create(&detect_thread, 0,
    detect_in_thread, 0)) error("Thread creation failed");
                    if(pthread_create(&bg_sub_thread, 0,
    bg_sub_in_thread, 0)) error("Thread creation failed");
               }


                //store number of people
                printf("the number of people:%d\n",countp);
                countpv[count]=countp;
                fprintf(output,"%d\n",countp);
                // show the total area of occupants
                printf("the total area of people:%d\n",total_area);


                if(!prefix){
                    show_image(disp, "Demo");
                    int c = cvWaitKey(1);
                    if (c == 10){
                        if(frame_skip == 0) frame_skip = 60;
                        else if(frame_skip == 4) frame_skip = 0;
```

```
                else if(frame_skip == 60) frame_skip = 4;
                else frame_skip = 0;
            }
        }else{
            char buff[256];
            sprintf(buff, "%s_%08d", prefix, count);
            save_image(disp, buff);
        }

            pthread_join(fetch_thread, 0);
        if(count%2==0){
            pthread_join(detect_thread, 0);
            pthread_join(bg_sub_thread,0);
        }



        if(delay == 0){
            free_image(disp);
            disp  = det;
        }
        det   = in;
        det_s = in_s;
    }else {
        fetch_in_thread(0);
        det   = in;
        det_s = in_s;
        detect_in_thread(0);
        if(delay == 0) {
            free_image(disp);
            disp = det;
        }
        show_image(disp, "Demo");
        cvWaitKey(1);
    }
    --delay;
    if(delay < 0){
        delay = frame_skip;


        double after = get_wall_time();
        float curr = 1./(after - before);
        fps = curr;
        before = after;
    }
```

```
        }
        free(countpv);
    }
    #else
    void demo(char *cfgfile, char *weightfile, float thresh, int
    cam_index, const char *filename, char **names, int classes, int
    frame_skip, char *prefix, float hier_thresh)
    {
        fprintf(stderr, "Demo needs OpenCV for webcam images.\n");
    }
    #endif
```

Appendix B: PMV-based HVAC control

- comppmv

```
function pmv = comppmv(ta,tr,vel,rh,met,clo,wme)
% returns [pmv,ppd]
% ta,air temperature
% tr, mean radiant temperature(C)
% vel, relative air velocity(m/s)
% rh, relative humidity(%)
% met, metabolic rate(met)
% clo, clothing(clo)
% wme, external work(met)
pa = rh * 10 * exp(16.6536 - 4030.183 / (ta + 235));

    icl = 0.155 * clo;  % thermal insulation of the clothing in M2K/W
    m = met * 58.15;  % metabolic rate in W/M2
    w = wme * 58.15;  % external work in W/M2
    mw = m - w;  % internal heat production in the human body
    if (icl <= 0.078)
        fcl = 1 + (1.29 * icl);
    else
        fcl = 1.05 + (0.645 * icl);
    end

    % heat transf. coeff. by forced convection
    hcf = 12.1 * sqrt(vel);
    taa = ta + 273;
    tra = tr + 273;
    tcla = taa + (35.5 - ta) / (3.5 * icl + 0.1);

    p1 = icl * fcl;
    p2 = p1 * 3.96;
    p3 = p1 * 100;
```

```
p4 = p1 * taa;
p5 = (308.7 - 0.028 * mw) + (p2 * (tra/100)^ 4);
xn = tcla / 100;
xf = tcla / 50;
eps = 0.00015;

n = 0;
while abs(xn - xf) > eps
    xf = (xf + xn) / 2;
    hcn = 2.38 * abs(100.0 * xf - taa)^0.25;
    if (hcf > hcn)
        hc = hcf;
    else
        hc = hcn;
    end
    xn = (p5 + p4 * hc - p2*xf^4) / (100 + p3 * hc);
    n = n + 1;
    if (n > 150)
        error('Max iterations exceeded');
    end
end


tcl = 100 * xn - 273;

% heat loss diff. through skin
hl1 = 3.05 * 0.001 * (5733 - (6.99 * mw) - pa);
% heat loss by sweating
if mw > 58.15
    hl2 = 0.42 * (mw - 58.15);
else
    hl2 = 0;
end
% latent respiration heat loss
hl3 = 1.7 * 0.00001 * m * (5867 - pa);
% dry respiration heat loss
hl4 = 0.0014 * m * (34 - ta);
% heat loss by radiation
hl5 = 3.96 * fcl * (xn^4 - (tra/100)^4);
% heat loss by convection
hl6 = fcl * hc * (tcl - ta);

ts = 0.303 * exp(-0.036 * m) + 0.028;
pmv = ts * (mw - hl1 - hl2 - hl3 - hl4 - hl5 - hl6);
```

```
    %{
    ppd = 100.0 - 95.0 * math.exp(-0.03353 * pow(pmv, 4.0)
        - 0.2179 * pow(pmv, 2.0))
    %}
```

- compheatsetp

```
function heat_setpoint = compheatsetp(temp0,vel,rh,met,clo,wme)
%calculate heating setpoint when occupied
%compheatsetp(21,0.1,55,1.0684,1,0)
%due to the incremental value, sometimes it may jump up and down and
never
%reach the comfort range...
pmv = comppmv(temp0,temp0,vel,rh,met,clo,wme);
pmvup = comppmv(temp0-0.1,temp0-0.1,vel,rh,met,clo,wme);
pmvdown = comppmv(temp0+0.1,temp0+0.1,vel,rh,met,clo,wme);

if (pmv <= -0.49 && pmv >= -0.51) || (pmvup <= -0.51 && pmvdown >= -
0.49)
    heat_setpoint = temp0;
elseif pmv >= -0.49
    heat_setpoint = compheatsetp(temp0-0.1,vel,rh,met,clo,wme);
else
    heat_setpoint = compheatsetp(temp0+0.1,vel,rh,met,clo,wme);
end
```

- compcoolsetp

```
function heat_setpoint = compheatsetp(temp0,vel,rh,met,clo,wme)
%calculate heating setpoint when occupied
%compheatsetp(21,0.1,55,1.0684,1,0)
%due to the incremental value, sometimes it may jump up and down and
never
%reach the comfort range...
pmv = comppmv(temp0,temp0,vel,rh,met,clo,wme);
pmvup = comppmv(temp0-0.1,temp0-0.1,vel,rh,met,clo,wme);
pmvdown = comppmv(temp0+0.1,temp0+0.1,vel,rh,met,clo,wme);

if (pmv <= -0.49 && pmv >= -0.51) || (pmvup <= -0.51 && pmvdown >= -
0.49)
    heat_setpoint = temp0;
elseif pmv >= -0.49
    heat_setpoint = compheatsetp(temp0-0.1,vel,rh,met,clo,wme);
else
    heat_setpoint = compheatsetp(temp0+0.1,vel,rh,met,clo,wme);
end
```

- coolsetp

```
load('occupant4.csv'); %*
```

```
load('activity_level.csv'); %**
load('clothing_insulation.csv'); %**
temp0=24;
set_back=26.7;
vel=0.2;
rh = 60;
wme=0;
cool_setpoint = occupant4;
cool_setpoint_act = occupant4;
ind0 = find(cool_setpoint==0);
ind1 = find(cool_setpoint>0);
cool_setpoint(ind0)=set_back;
cool_setpoint_act(ind0)=set_back;
for i = 1:length(ind1)
    id = i
    met=activity_level(ind1(i));
    clo=clothing_insulation(ind1(i));
    clo_act = clothing_activity(ind1(i));
    cool_setpoint(ind1(i))=compcoolsetp(temp0,vel,rh,met,clo,wme);

cool_setpoint_act(ind1(i))=compcoolsetp(temp0,vel,rh,met,clo_act,wme)
;
end
csvwrite('cooling_setpoint.csv',cool_setpoint);
csvwrite('cooling_setpoint_activity.csv',cool_setpoint_act);
```

% *occupant4.csv is generated from occupancy simulator whose ID: 2813712

% **activity_level.csv and cloting_insulation.csv are generated according to the approach described in section **Methodology**

- heatsetp

```
load('occupant4.csv');
load('activity_level.csv');
load('clothing_insulation.csv');
temp0=25;
set_back=15.6;
vel=0.2;
rh = 60;
wme=0;
heat_setpoint = occupant4;
heat_setpoint_act = occupant4;
ind0 = find(heat_setpoint==0);
ind1 = find(heat_setpoint>0);
heat_setpoint(ind0)=set_back;
heat_setpoint_act(ind0)=set_back;
```

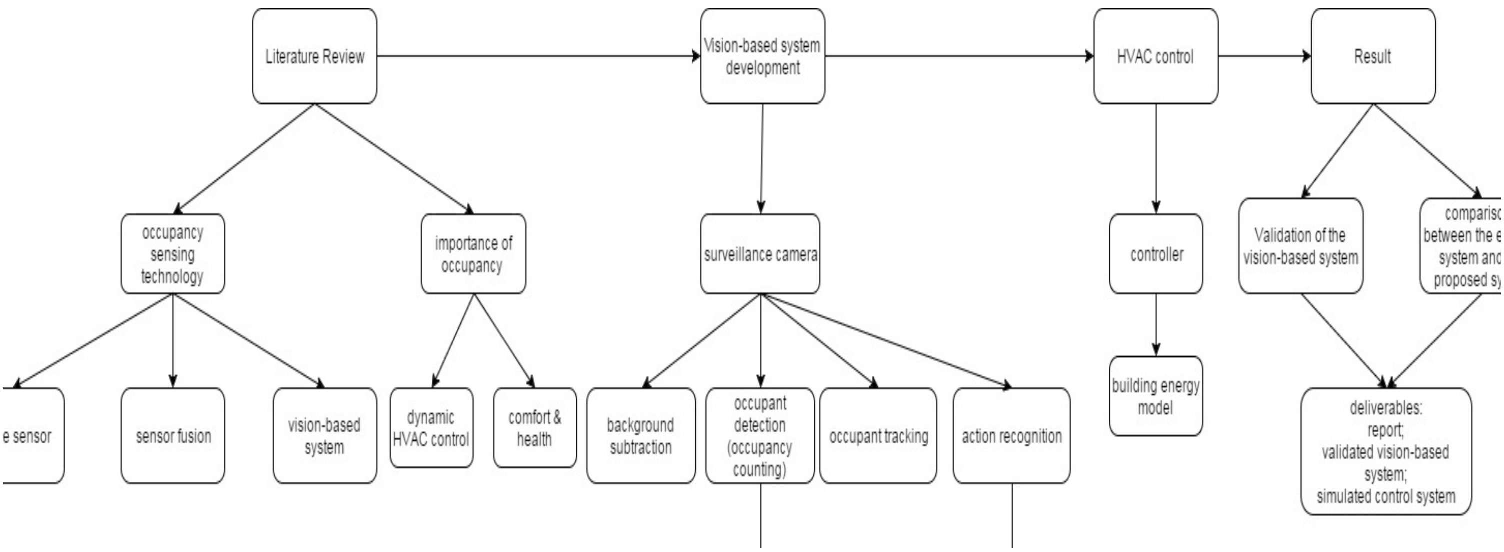
```
for i = 1:length(ind1)
    id = i
    met=activity_level(ind1(i));
    clo=clothing_insulation(ind1(i));
    clo_act = clothing_activity(ind1(i));
    heat_setpoint(ind1(i))=compheatsetp(temp0,vel,rh,met,clo,wme);

heat_setpoint_act(ind1(i))=compheatsetp(temp0,vel,rh,met,clo_act,wme)
;
end
csvwrite('heating_setpoint.csv',heat_setpoint);
csvwrite('heating_setpoint_activity.csv',heat_setpoint_act);
```

- final

```
load('cooling_setpoint.csv');
load('heating_setpoint.csv');
cooling_setpoint(cooling_setpoint>26.7)= 26.7;
heating_setpoint(heating_setpoint<15.6)=15.6;
difference=cooling_setpoint - heating_setpoint;
csvwrite('cooling_setpoint_final.csv',cooling_setpoint);
csvwrite('heating_setpoint_final.csv',heating_setpoint);
```

Appendix C: Flow Chart of Methodology



Appendix D: Occupancy detection with Darknet

CARNEGIE MELLON UNIVERSITY

School of Architecture
College of Fine Arts

Thesis

Submitted in Partial Fulfillment of the requirements for the degree of

Master of Science in Building Performance and Diagnostics

TITLE:

Dynamic HVAC operations based on occupancy patterns with real-time vision-based system

AUTHOR:

Siiang Lu

ACCEPTED BY ADVISORY COMMITTEE:

ERICA COCHRAN HAMEEN, ______ April 18, 2018
Professor NAME Principal Advisor DATE

______ ______
Professor NAME Advisor DATE

______ ______
Professor NAME Advisor DATE

______ ______
Professor NAME Advisor DATE