

**EVENT DETECTION FOR SMART CONFERENCE ROOM  
USING MULTI-STREAM CONVOLUTIONAL NEURAL NETWORK**

**BY**

**BELINDA KHOO PAI LIN**

**A REPORT**

**SUBMITTED TO**

**Universiti Tunku Abdul Rahman**

**in partial fulfillment of the requirements**

**for the degree of**

**BACHELOR OF COMPUTER SCIENCE (HONS)**

**Faculty of Information and Communication Technology**

**(Kampar Campus)**

**JAN 2020**



## REPORT STATUS DECLARATION FORM

**Title:** EVENT DETECTION FOR SMART CONFERENCE  
ROOM USING MULTI-STREAM CONVOLUTIONAL  
NEURAL NETWORK

**Academic Session:** JAN 2020

I BELINDA KHOO PAI LIN  
**(CAPITAL LETTER)**

declare that I allow this Final Year Project Report to be kept in

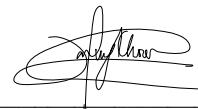
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,



(Author's signature)



(Supervisor's signature)

**Address:**

21, LORONG KURAU 1,  
CHAI LENG PARK,  
13700 PERAI, PENANG.

Tan Hung Khoon

Supervisor's name

**Date:** 22 APRIL 2020

**Date:** 24 April 2020

**EVENT DETECTION FOR SMART CONFERENCE ROOM  
USING MULTI-STREAM CONVOLUTIONAL NEURAL NETWORK**

**BY**

**BELINDA KHOO PAI LIN**

**A REPORT**

**SUBMITTED TO**

**Universiti Tunku Abdul Rahman**

**in partial fulfillment of the requirements**

**for the degree of**

**BACHELOR OF COMPUTER SCIENCE (HONS)**

**Faculty of Information and Communication Technology**

**(Kampar Campus)**

**JAN 2020**

## DECLARATION OF ORIGINALITY

I declare that this report entitled “**EVENT DETECTION FOR SMART CONFERENCE ROOM USING MULTI-STREAM CONVOLUTIONAL NEURAL NETWORK**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature :  \_\_\_\_\_

Name : BELINDA KHOO PAI LIN

Date : 22 APRIL 2020

## ACKNOWLEDGEMENTS

I would like to express my sincere gratitude and appreciation to my supervisor, Dr. Tan Hung Khoon who gave me the golden opportunity to engage in this deep learning project on the topic “**EVENT DETECTION FOR SMART CONFERENCE ROOM USING MULTI-STREAM CONVOLUTIONAL NEURAL NETWORK**”, which also helped me on doing a lot of researches and also guided me well throughout this project. It is the first step to establish my career in the field of deep learning.

Secondly, I would like to express my deep appreciation to Dr. Tan Hung Khoon and Ms Lai Siew Cheng for providing assistance and computational resources throughout the development process of this project. Without these resources, the project would not be completed on time.

Thirdly, I would like to thank my parents and my family for their unconditional love, support and continuous encouragement throughout the project.

Lastly, I would like to say thank you to a very responsible and collaborative person, Tan Yi Jian, who is also my project partner, for his patience, unconditional supports and cooperation throughout this project.

## ABSTRACT

Conferencing/meeting is an activity that is inevitable in almost every workplace because it acts as a prominent role for determining the future of business operations. Hence, meeting rooms are designed as a space to accommodate various meeting activities especially for effective decision-making. In order for companies to effectively manage countless of meetings per day, various systems are being developed as an aid in the meeting room. In fact, most of the existing systems are developed upon occupancy analysis technique, which are just aimed to detect the presence of occupants in a meeting room instead of the on-going activities. Event detection in meeting rooms is critically important as one may misuse the conference room by occupying it merely for irrelevant purposes. To ensure that everyone is utilizing company resources in a proper way, this project delivers a web-based Smart Conference Room System for classifying the happening events in meeting rooms. In order to achieve this, some human action recognition techniques would be applied for capturing and understanding the motion information of the occupants.

In this project, the R(2+1)D with variant of 34 layers architecture (Tran et al. 2018) is proposed as the network architecture and it will be built within a two-stream framework for capturing the spatiotemporal features of a video. The model is pretrained on the Kinetics Human Action dataset before finetuning with the Conference Dataset collected from meeting rooms in company X. The raw footages collected from Company X are being preprocessed through in-depth data annotation and labelling based on the on-going activities in different meeting rooms.

After the successful attempt of acquiring the pretrained model, the learned features and weights are then transferred for finetuning the newer model that is based on the preprocessed Conference Dataset. Consequently, the newer model is integrated into a web-based system in order to handle event detections in a meeting room. Apart from that, one of the approaches for object detection, You Live Only Once, also known as YOLO, will be incorporated into this system to act as an object counter for providing extensive information. Additional analytics are delivered in this system for companies to gain insights into the usage of meeting rooms.

# TABLE OF CONTENTS

<b>TITLE PAGE</b>	<b>i</b>
<b>DECLARATION OF ORIGINALITY</b>	<b>ii</b>
<b>ACKNOWLEDGEMENTS</b>	<b>iii</b>
<b>ABSTRACT</b>	<b>iv</b>
<b>TABLE OF CONTENTS</b>	<b>v</b>
<b>LIST OF FIGURES</b>	<b>vii</b>
<b>LIST OF TABLES</b>	<b>ix</b>
<b>LIST OF ABBREVIATIONS</b>	<b>x</b>
<b>CHAPTER 1 INTRODUCTION</b>	<b>1</b>
1.1 Project Background	1
1.2 Meeting Event Detection for a Smarter Meeting Room	3
1.3 Project Scope and Objectives	6
1.4 Work Distribution	7
<b>CHAPTER 2 LITERATURE REVIEW</b>	<b>9</b>
2.1 Smart Conference Room Systems	9
2.1.1 Alexa for Business	9
2.1.2 Wireless IoT-based Occupant Detection in Smart Conference Room	11
2.2 Action Recognition in Videos	13
2.2.1 Hand-crafted Methods	13
2.2.2 Deep Learning Systems	16
2.3 Object Detection	23
<b>CHAPTER 3 MEETING ROOM EVENT DETECTION</b>	<b>27</b>
3.1 System Overview	27
3.2 Event Detection for SCR	29
3.2.1 Targeted Events	29
3.2.2 Proposed Event Detection Method	30
3.3 Dataset Generation and Preparation	33
3.4 Experimental setup	35
3.5 Experiments	35
3.5.1 RGB Network Configuration(s)	35
3.5.2 OF Network Configuration(s)	38
3.5.3 Fused Two-Stream Network	39

<b>CHAPTER 4 WEB APPLICATION DEVELOPMENT</b>	<b>41</b>
<b>CHAPTER 5 CONCLUSION</b>	<b>46</b>
<b>BIBLIOGRAPHY</b>	<b>47</b>

# LIST OF FIGURES

Figure Number	Title	Page
Figure 1.1	Empty meeting room	4
Figure 1.2	Meeting is on going	4
Figure 1.3	Room hogging	5
Figure 2.1	Event flow of "Alexa for Business" (Amazon Alexa 2017)	9
Figure 2.2	Overview of deployed IoT sensing equipment (Saralegui et al. 2019)	12
Figure 2.3	Visualization of candidate matches of homography estimation (Wang & Schmid 2017)	15
Figure 2.4	Estimated homography with and without human detectors	15
Figure 2.5	C3D architecture (Tran et al. 2015)	17
Figure 2.6	Visualization of C3D, using the method of deconvolution (Tran et al. 2015)	18
Figure 2.7	Inflated Inception-v1 on the left and its detailed inception submodule on the right (Carreira & Zisserman 2017)	19
Figure 2.8	Two-Stream I3D architecture (Carreira & Zisserman 2017)	20
Figure 2.9	(2+1)D convolution (Tran et al. 2018)	21
Figure 2.10	R(2+1)D which are ResNets with (2+1)D Convolutions	22
Figure 2.11	Two-stream architecture in video classification (Simonyan & Zisserman 2014)	23
Figure 2.12	Bounding boxes with location prediction and dimension priors (Redmon et al. 2016)	25
Figure 2.13	Darknet-53 (Redmon et al. 2016)	26
Figure 3.1	Overview of system	27
Figure 3.2	Overview of proposed model in SCR module	30
Figure 3.3	RGB, OF-Horizontal, OF-Vertical frames of "Meeting".	31
Figure 3.4	R(2+1)D-34 architecture (Tran et al. 2018)	32
Figure 3.5	R(2+1)D with 34 layers specifications	32
Figure 3.6	Training accuracy, training loss and stepLR scheduler	36
Figure 3.7	Comparison between two LR schedulers	37
Figure 4.1	Login page	41

Figure 4.2 Administration page (Index page)	42
Figure 4.3 'Meeting' in room A	42
Figure 4.4 'Non-Meeting' in room B	43
Figure 4.5 'Empty' in room C	43
Figure 4.6 Data Analytics Page	44
Figure 4.7 Developer Page	45

## LIST OF TABLES

<b>Table Number</b>	<b>Title</b>	<b>Page</b>
Table 1.1	Tasks distribution	8
Table 3.1	Time taken for training process	35
Table 3.2	Results of RGB network with different settings	37
Table 3.3	Results of OF network with different settings	39
Table 3.4	Final performance of RGB, OF and Fused stream	39

## LIST OF ABBREVIATIONS

<i>API</i>	Application Program Interface
<i>CNN / ConvNet</i>	Convolutional Neural Network
<i>CO<sup>2</sup></i>	Carbon Dioxide
<i>EWMA</i>	Exponentially Weighted Moving Average
<i>FC</i>	Fully-Connected
<i>GPU</i>	Graphics Processing Unit
<i>HVAC</i>	Heating, Ventilation and Air-Conditioning
<i>GUI</i>	Graphical User Interface
<i>IoT</i>	Internet of Things
<i>LSTM</i>	Long Short-Term Memory
<i>LR</i>	Learning Rate
<i>MPC</i>	Model Predictive Control
<i>NLP</i>	Natural Language Processing
<i>OF</i>	Optical Flow
<i>PIR</i>	Passive Infrared Sensors
<i>ReLU</i>	Rectified Linear Unit
<i>RGB</i>	Red-Green-Blue (Channels)
<i>ROI</i>	Region of Interest
<i>SCR</i>	Smart Conference Room
<i>SGD</i>	Stochastic Gradient Descent

*SVM*

Support Vector Machine

## CHAPTER 1 INTRODUCTION

### 1.1 Project Background

As of today, conferencing is definitely a recrudescing pain point in the workspace environments. From the way of defining conference goals, gathering participants, scheduling an agenda until reserving a conference room, people would struggle a lot for hosting a successful conference. Nevertheless, conferencing carries the utmost responsibility for bringing people together to achieve effective communication and boost up productivity, which is inevitable in every workplace. Thus, a conference room is designed to keep highly sensitive information and conversation in dedicated spaces. For instance, new product launches, breakthrough ideas, innovations, solutions and others. The designated spaces also form great atmosphere for companies to welcome clients in addition to build up their confidence towards the companies. Furthermore, conference room is the best place to unite teams of people in order to foster quality connections among team members and to carry out effective brainstorming sessions. It not only acts as a think tank but also creates excellent atmosphere for encouraging people to speak out their minds frankly and publicly. In short, conference room is a significant key to determine the futures of an organization.

Nevertheless, it is undeniably that no matter how many conference rooms there are in an office or a company, they seem to get reserved mysteriously. Is the company have underestimated the demand for conference rooms and built too few of them? Or, is this a clear sign indicating that the management of meeting rooms is needed to be prioritized in order for catering more meeting activities? When the number of conference rooms are insufficient to accommodate more effective meetings, frustrating situations like tense working environment may be induced due to low productivity of businesses.

On top of that, awkward situations may be encountered especially in multinational corporations, as there would be high demand for conference rooms during several popular time slots, which is caused by the geological differences. There also comes a time when a pre-booked conference room has been occupied by other people, which is an unpleasant scenario. For instance, if two teams want to jump in an empty conference room at the same time without prior reservation, a workplace conflict may be induced. Despite that majority of organizations have deployed scheduling systems which smoothen the reservation procedures of meeting rooms, there might be also some

“squatters” who inappropriately monopolizes meeting room alone for long periods, just to work and rest quietly or to conduct one-to-one meeting at the expense of group needs. Moreover, in order to save time, people might reserve conference rooms for consecutive weeks or months in advance just to pre-occupy the rooms for recurring meetings. As soon as the space is being reserved by someone, it is then off-limits to all other users. This may lead to wastage of company resources if the reserved rooms are cancelled without releasing the rooms.

Perhaps some of the employees would utilize large spaces just to conduct a small-grouped meeting. For instance, a meeting room built for 30 people is being utilized by only a team of 3 people. In addition, there might be circumstances where people reserve teleconference room facilities just for on-site meetings. All of these could lead to an inefficient usage of company resources. Furthermore, power wastage could take place if the Heating, Ventilation and Air-Conditioning (HVAC) systems are not turned off appropriately after meetings. As such, the productivity of businesses could be diminished as other project teams are unable to have opportunities to utilize the meeting rooms for facilitating team collaborations. Hence, company resources should be managed and monitored in an effective and efficient manner.

As the world becomes more and more digital, a number of applications such as face recognition systems, object detection, augmented reality and video surveillance system has been introduced to the world for improving life quality. In order to conduct a more effective and efficient meetings, different types of smart conference room technologies have been launched to foster collaborations among employees. To illustrate, an artificially intelligent speaker that is built on top of NLP, is designed to streamline meetings in a meeting / conference room. Sitting at the middle of a meeting table, the speaker will respond to individuals who provide instructions via voice commands. The audio inputs received will first be decoded into a series of tasks via a speech recognition model. After interpreting the messages, the system would then perform the tasks correspondingly without any delay. As such, a smarter workplace could be created by simplifying meeting room experiences for employees.

It is undeniable that the building energy use is mainly dominated by heating, cooling and lighting, whereby lighting dominates the most in terms of energy usage. Thus, a smart conference room with the application of lighting control system (Afshari et al.

2015) is established for brightness controls, vacancy detection and personal controls. The system could automatically activate different lights based on acoustic events identified in the room, which then reduces energy costs in the enterprises.

Apart from that, there are some smart meeting room systems, whereby multiple type of sensors is used to monitor room occupancy (Saralegui et al. 2019). As such, the issue of “phantom” booking could be eradicated as well as if no one is being detected within certain minutes after the scheduled time, the system will automatically update the room status, making it accessible to other employees. However, in spite of scheduling and managing meeting room resources, these existing smart conference room systems are incapable of identifying human activities on-the-go in the meeting rooms. Thus, companies are unable to ensure that all the rooms are being utilized appropriately with proper use cases.

Having discovered why meeting rooms are often the go-to place for many employees, it actually could be deduced by saying that the insufficient number of meeting rooms is not the key of these problems. In order to optimize the meeting room usage, a piece of smart conference room system is delivered in this project for identifying employees’ activities in conference rooms based on visual analytics gathered via a surveillance camera. As such, companies are able to ensure that all the meeting rooms are being used with proper and intended use cases. Thus, the utilization of conference spaces in companies could be maximized as meetings are crucial for driving the consistency of businesses’ success.

## **1.2 Meeting Event Detection for a Smarter Meeting Room**

Despite that there are already numerous systems built and integrated for SCR, there are still limited attempts on utilizing human action recognition task in conferencing environment. Hence, a smart conference room with the integration of human action recognition technique, is proposed in this work in order to detect and recognize the on-going activities in meeting rooms. Before diving into detailed information of this system, the concept of human action recognition is discussed in the first place.

Human action recognition (HAR) is a technique that automatically identifies human movements based on its interactions with the environment. Every human action is done for intended purposes. For example, while walking in a street, the person will interact

with the street using his / her arms, hands, legs, bodies and so on. Nevertheless, it is often being considered as the most challenging classification task as it requires to predict the movement of a person based on multiple streams of complex and expensive data that are collected from massive amounts of photos and videos. In order to achieve such a difficult task, several contexts should be considered including emotions, gestures, actions, physical environments and so on.

By developing a SCR system incorporated with techniques of human action recognition, several events could be detected via visual inputs from a surveillance camera. As such, this system is capable of indicating whether the rooms are vacant or occupied, and the occupants are using the room appropriately or inappropriately. In other words, the system will show no response if there is no any occupant detected in the room. However, if there are participants in the room, it would identify whether the person is actually having a meeting or not. If the person is found to be hogging the room for privacy usage, the system might automatically turn off the HVAC system as an alert to restrict the person from using the room at that moment. The sample images for each interesting event are listed in the figure below.



**Figure 1.1 Empty meeting room**



**Figure 1.2 Meeting is on going**



**Figure 1.3 Room hogging**

By detecting the presence of occupants and the activities they are carrying out in the meeting room, different types of meeting analytics could be generated. Precise analytics such as the usage rate of the rooms along with the number of presenters and chairs in the meeting room could be acquired. Such information is essential for companies so that better policies could be imposed for managing and scheduling the meeting venues more efficiently. As such, company could utilize these analytics to identify the needs for meeting room and to decide if it is necessary to expand the number of meeting rooms or to devise more efficient scheduling systems for tackling the problems.

Different departments may have different requirements on utilizing the meeting rooms. For example, IT department may necessitate high-tech equipment to demonstrate the prototypes developed, and may use the room for longer time due to the nature of business meeting, e.g., the refinement process of solution requirements. By acquiring analytics via the system, company could understand how frequent and how long the meeting rooms are being utilized by each department. Apart from that, cases of inappropriate usage such as vandalism or room hogging could be identified and reported to the management team. By tracing the records from surveillance camera, the people found to be abusing the rooms could be penalized and punished by the management team. As such, companies are able to ensure that the meeting rooms are utilized properly and well-maintained.

Furthermore, this system could also be integrated with lighting control system for brightness controls, vacancy detection and personal controls. Instead of spontaneously turning on and off the lights based on the presence of occupants, it could also control the lighting system to activate different lights based on events identified in the room, which then minimizes energy costs in the enterprises. For example, when a person is

making presentation, the surrounding lights should be dimmer. Consequently, the efficiency in management of the workplaces could be enhanced in order to achieve the company goals along with low energy cost, which then leads to the successful future of a company.

### **1.3 Project Scope and Objectives**

In order to tackle these issues aforementioned, a piece of real-time web-based smart conference room system is delivered for event detections in a conference room. While monitoring human actions, the system can even integrate with other system in order to control lightings of conference rooms and to keep the office scheduling system up-to-date at all times. In light of fact, effective meetings are one of the keys that drives the consistency of businesses' success. By developing a SCR system integrated with techniques of action recognition and object detection, the utilization of conference spaces in a company could be optimized as much as possible, which then improves the business operations and productivities.

The proposed system comprises of 3 objectives, which are:

#### **I. A deep learning system for detecting meeting events in conference videos.**

This project aims to detect and classify human activities in conference rooms as well as to ensure that the rooms are being used with proper use cases. Only several actions such as meeting, presenting and typing on keyboard are considered as meeting activities, or else idle activities in the rooms such as room hogging and engaging in a phone call are considered as non-meeting activities. In order to build a system integrated with human action recognition, deep models such as ConvNets are required for performing video analysis and making predictions. In this case, the video inputs will be convolved over multiple CNNs in order to extract spatial and motion features for on-going event detections in conference rooms. The implementation details will be discussed later.

#### **II. A new dataset for training meeting event detection.**

To accomplish this project, huge datasets are required to train models and obtain justifiable results. Nevertheless, the amount of resources for this type of training data is still restricted due to the nature of businesses since the conferencing videos often contain information that is private and confidential to the organization. Besides,

footages provided by companies could be blurry and in poor quality, causing loss of detailed information. Since installing and running a high-quality surveillance camera are costly, companies usually opt for the one with only average quality as it is cheaper. The way of locating surveillance cameras in conferencing environment might influence video quality too. Owing to the fact that majority of companies and employers tend to place the surveillance cameras up high on walls or ceilings, these cameras have to record videos from distant and widen areas, causing the targeted people to be captured in small-sized. Thus, datasets are needed to be prepared and preprocessed properly at the beginning of the project development.

### III. Data analytics for smart conference room.

By enhancing the competence of this system, object detection technique would be implemented to act as an object counter for people and chairs. As such, it could aid in producing analytics for companies to identify the demands for meeting rooms. For example, precise analytical information such as the occupancy rate of each meeting rooms could be provided since the number of occupants and seats in a conference room could be monitored from time to time. This information is vital for deducing if the conference room is abused or underused. By doing this, the effectiveness in managing company resources, such as meeting rooms, might be revamped by using the inferences of room usage drawn from the information gathered in this system.

#### 1.4 Work Distribution

It is an industrial project with a multi-national company. It is a joint project of two members whereby the other group member is Tan Yi Jian. Owing to the reason that this project is conducted within a group of two persons, a series of tasks are discussed and distributed in a fair and equal manner, whereby each of us shared the equivalent amount of contributions to this project. The distributed tasks are as follow:

No.	Task(s)	Member
1.	Data preparations including data cleaning, data annotations and data preprocessing (200 hours of footages)	Belinda, Yi Jian (100-hours-footages for each person)
2.	Implementation of optical flow stream within two-stream framework	Belinda

3.	Implementation of RGB stream within two-stream framework	Yi Jian
4.	Fusion of two-stream network	Belinda, Yi Jian
5.	Implementation of YOLOv3 as an object counter	Belinda, Yi Jian
6.	Web application (user interfaces) for system administrator mode	Belinda
7.	Web application (user interfaces) for developer mode	Yi Jian

**Table 1.1 Tasks distribution**

## CHAPTER 2 LITERATURE REVIEW

Several types of SCR Systems are introduced to resolve the issues stated above. The features of SCR Systems include automated lighting control, voice assistant and so on. This chapter will be divided into 3 sections, which are Section 2.1, 2.2 and 2.3 respectively to discuss about the existing SCR systems, techniques used for implementing Human Action Recognition and Object Detection.

### 2.1 Smart Conference Room Systems

In this section, a product review on two commercial SCR Systems is performed. In Section 2.1.1, an NLP-based smart conference system called Alexa is introduced. Then, Section 2.1.2 reviews an IoT-based conference system which uses an array of sensors to detect meeting activities.

#### 2.1.1 Alexa for Business

Alexa for Business (Amazon Alexa 2017) is a service provided by Amazon to improve businesses' daily operations and increase productivities in the workplaces. It is built based on Natural Language Processing, usually known as NLP, which deals with the interactions between human natural languages and computers. To illustrate, when users make requests from Alexa, it records and sends to Amazon's server in order to find the corresponding words. After interpreting the requests from information gathered, Alexa then responds to the users by speaking or performing the requested activities.



**Figure 2.1 Event flow of "Alexa for Business" (Amazon Alexa 2017)**

Amazon's Alexa provides many intelligent features based on NLP technologies, which is considered a new and excellent platform to bring voice interactions to businesses and

corporations. First of all, Alexa allows a meeting room to be voiced control where users issue instructions, e.g., turning on and off devices, by means of voice commands. A proprietary speech recognition model is used to decode voice commands into built-in functions. The model is built on the Amazon's cloud platform and is relying on their device, Amazon Echo speakers, for streaming audio inputs from users for further processing. As such, it effectively eliminates all the necessities for users to manually handle the equipment in a conference room. With Alexa for Business, working operations can be made much more efficiently using voice commands regardless of where the employees are in the workplace. It eventually builds a smarter workplace which simplifies meeting room experiences for the employees.

Apart from that, Alexa supports customized voice commands based on the need of individual users through a development kit. Using the development kit, people are allowed to build their personalized functions for enhancement of device capabilities as well as for richer user experiences. For example, while integrating with personal cloud calendar, Alexa could be utilized as an intelligent voice assistant by allowing users to manage personal schedules, update to-do lists and set reminders by means of voice commands. This could be achieved as Alexa is built on top of cloud-based intelligence where natural language understanding, text-to-speech processes and complex automatic speech recognition are impressively handled in the platform. Thus, this aids users in streamlining their tasks with a faster and simpler way.

It is undeniably that Alexa benefits a great number of companies owing to its outstanding competences in handling voice commands. However, data privacy and security issues are always the major concerns due to the always-on microphone associated. Collections of personal information would be processed and stored in Amazon cloud regularly as long as the microphone is turned on. Thus, people may gain access to the information and monetize it in the form of targeted advertisements. In addition, Alexa is unable to handle any visual inputs. For instance, Alexa is incapable of recognizing face, detecting number of people and monitoring their activities in the conference room. Hence, it is unable to identify on-going events in the area. It only works from a general perspective which is to help users in solving problems through the standard voice commands from users. As a result, it is unable to react appropriately according to the surrounding situation of the conference room.

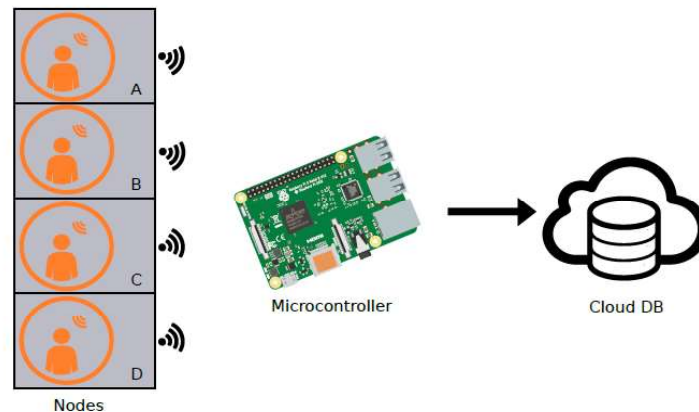
### 2.1.2 Wireless IoT-based Occupant Detection in Smart Conference Room

Recently, Internet of Things (IoT) devices are getting famous in the aspects of sensing and actuating. They are enhanced with the capabilities to converse with one another and perform different tasks for themselves via the connection to the local network. One of the fields which has acquired great impacts from IoT is the smart building system, owing to the reason that most people spend their time inside a building. For instance, their home, restaurant, office, library and so on. Equipping with various sensors and connecting up with multiple objects, these devices are able to collect real-time information in certain environment and perform certain tasks in a timely manner. To illustrate, for smart home or conference room, the information can be used to regulate the air-conditioning temperature and turn on / off electrical devices.

As buildings are one of the entities that consuming high power resources, it is a necessity to deploy a power monitoring system in buildings to avoid power wastage. On top of that, it is proven that human behaviour is the main dominance of energy usage in a building. For example, turning on / off lights or air-conditioner. Thus, to reduce the overall energy consumption of buildings, (Saralegui et al. 2019) proposed a system which is based on a technique namely Model Predictive Control (MPC). The predictive models are developed to control HVAC systems while satisfying the requirements and occasions happened in a real building. A wireless interconnected sensing network is deployed in four meeting rooms in a company for monitoring human presence and changes in thermal behaviors. The network is based on six different types of sensors such as relative humidity, temperature, vibration, ultra-violet radiation, motion and light sensors. These sensors are installed in each meeting room for detecting changes and transferring data collected among each other. Later, the information gathered would be extracted into usage patterns and would be utilized for posterior predictions of room usage.

Specifically, four of the connected nodes in this network would aggregate their data and establish communication among each other via a microcontroller device. The microcontroller device collects all the data from every room via wireless communication and explicitly stores them into a database so that people can access the data via API. The information obtained will be fed to Model Predictive Control strategies, so that it would be possible to predict future human room occupancy and

conserve energy by efficient lighting control. Besides, the data gathered can be used as room usage analytics in order to discover patterns, which could be useful in estimation of energy consumption as well.



**Figure 2.2 Overview of deployed IoT sensing equipment (Saralegui et al. 2019)**

Since the multi-sensors do not provide occupancy information directly, an automatic data modeling process is used to convert the raw data collected into binary occupancy data with an interval of 5 minutes. For every 30 minutes, multiple 5-minute information would be averaged to obtain more representative value in order to avoid misreading. For instance, misreading will occur when the occupant has no moving actions in the meeting room.

One of the main advantages of this system is that the efforts and costs in deploying such a complex network is significantly reduced by integrating wireless IoT equipment in the system. Only four nodes along with a microcontroller which is responsible for data collection, storage and data accessibility is used. This allows quick and easy installations which is such a huge improvement in terms of scalability and flexibility. Besides, this work completely eliminates the needs for people to explicitly interact with the system. The automation process can cut down the operational cost and energy. For instance, if there is no any human presence detected in a meeting room, the microcontroller will immediately switch off all the lights and equipment in the particular room, which eventually reduces the energy consumption.

The limitation of this work is that the system may fail in functioning correctly in the rooms that are larger than the detection range of the sensors. Despite the fact that this could be resolved by deploying more PIR sensors, it may become cost-ineffective and

may require more efforts for multiple sensors installation. Other than that, if the size of the room is not large enough, sensors like CO<sup>2</sup> sensors and sound are practically infeasible due to the reason that the measurement of sound and CO<sup>2</sup> may fluctuate wildly. As a consequence, the system may provide false positive or negative results. Furthermore, PIR sensors are only able to capture the presence of people instead of their actions. Thus, the system is unable to distinguish whether the rooms are utilized appropriately with proper use cases.

Based on the consideration of these issues, in order for a system to work more reliably, the system must be able to reflect on visual inputs. This is due to the reason that occupancy analysis-based systems are restricted to merely detect the occupancy conditions without considering the context of human activities. Hence, this project is developed based on a more robust technique such as human action recognition technique in order to detect event in conference room based on the motion information.

## **2.2 Action Recognition in Videos**

Automatic understanding of human activities and his/her interactions with the physical environment have been one of the popular research areas in recent years. Due to its wide variety of potential applications in the real-world environment, many researchers have worked on designing various kinds of deep learning and hand-crafted approaches in order to improve accuracy and performance for action recognition. Unlike image classification, the feature representation of human actions in video describes not only the human appearance the spatial form, but also motion among the image sequences which could only be extracted via temporal changes. Hence, tasks of interpreting what is happening in a video could be extremely challenging compared to image classification.

This section will be divided into 2 sub-sections where Section 2.2.1 describes one of the popular hand-crafted approaches in handling action recognition and Section 2.2.2 presents several deep learning systems including C3D, I3D and R(2+1)D network architecture, which are superior towards the field of video processing and recognition.

### **2.2.1 Hand-crafted Methods**

Before the deep learning era, action recognition in videos are mainly performed through traditional hand-crafted representation-based approach. These methods focus on local

feature or holistic feature detectors and descriptors which are being designed through experience. Generally, a hand-crafted method is made up of three main phases such as foreground detection, hand-crafted feature extractions and classification accordingly. After developing a feature descriptor based on self-experience, useful features are extracted from sequences of frames. Later, a common classifier, e.g., Support Vector Machine (SVM) is trained for performing classification. Some of the popular hand-crafted approaches on performing action classification are Scale Invariant Feature Transform (SIFT), Space Time Interest Point (STIP) and Improved Dense Trajectories (IDT). Since iDT (Wang & Schmid 2017) had achieved the state-of-the-art at the moment prior to the growth of deep neural networks, let's take a look into this work.

Owing to the tremendous expansion of the real-world video data in terms of its size and complexity, more and more challenges have gradually emerged in the field of video recognition. To illustrate, there might be diverse ways and time intervals required for different people to perform certain actions. This might lead to intra-class variations and it may be confusing for models in learning the features. Similar to image recognition, the problematic issues like background clutters and obstructions might still happen in video recognition. Moreover, the moving background objects which lead to camera motion and motion clutter, may result in the variability of learning patterns, which then turn down the performance in solving the tasks. Low quality of video data which is induced by noises from sensors, countless video decoding artifacts and camera jitter, might be one of the key challenges as well. Furthermore, high computational cost is required for solving such complicated tasks due to the heavy processing of large-scaled datasets.

In order to solve the issues aforementioned above, (Wang et al. 2013) proposed an approach based on dense trajectory features in order to densely sample the feature points on numerous spatial scales. Making it possible on tracking the feature points reliably, the points located in similar areas are suppressed, and this could be done by median filtering in a dense OF field introduced by (Farneback 2003). This approach is shown to be efficient in representing motions by tracking them based on the displacement information gained via dense OF algorithm. Subsequently, inspired by the achievement of dense sampling in handling motion information, (Wang & Schmid 2017) proposed a network architecture, iDT, which robustly estimates human motions. This could be achieved by tracking the dense points using displacement information

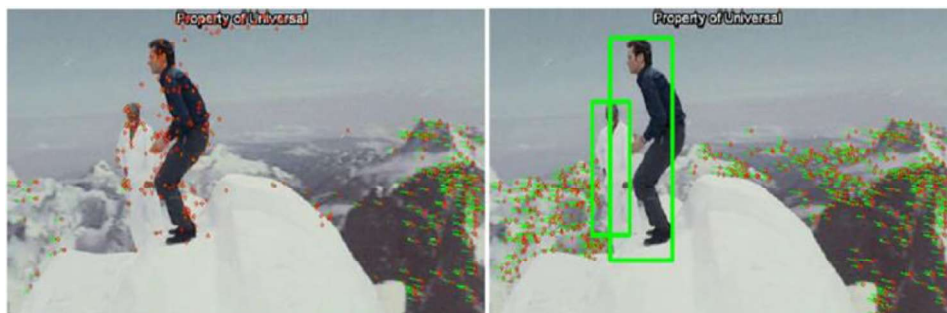
obtained from dense OF. The implementation of this work is almost comparable with the one mentioned earlier.

Additionally, an explicit camera motion estimation is established by extracting features point that matches between sequences of frames by using Farneback's dense optical flow and SURF descriptors. An adequate yet corresponding candidate matches are generated and used as homography estimation. Figure below shows the visualization of homography estimation. The red arrows correspond to dense optical flow matches while green arrows are from SURF descriptors.



**Figure 2.3 Visualization of candidate matches of homography estimation  
(Wang & Schmid 2017)**

Since human motion is not consistent with camera motion estimation, a robust human detector (Prest et al. 2012) which combines several part-based detectors, is applied in this work for extractions of different parts of human features. By doing so, the possible outlier matches could be eliminated during camera motion estimation, making the system to be more powerful. In other words, the global background motions could be eliminated as the background trajectories are already removed.



**Figure 2.4 Estimated homography with and without human detectors**

To sum up, the proposed system had significantly outperformed the state-of-the-art video representation and solved action recognitions task efficiently.

However, there might be several drawbacks for the traditional approaches in performing action recognitions. For most of the hand-crafted methods, they require an accurate human modeling and tracking, which still remains as a difficult problem up to this time. In addition, some of them are significantly computational expensive and time-consuming than modern techniques as feature detectors and descriptors are manually engineered by the researchers. If there are many classes of objects or events for detection and recognition, a great number of descriptors are needed to be designed for them one by one in order to extract diverse features, and this is impossible due to the time constraint. Thus, deep learning systems are introduced recently with the concept of end-to-end learning where machines would discover the underlying patterns of each specific class of objects automatically without any manual intervention.

### **2.2.2 Deep Learning Systems**

In recent years, the research community has significantly drawn lots of attentions to video analysis due to its complex structure. Unlike static image classifications, both spatial and temporal information are essential for video analysis and high accuracy action recognition. Temporal features extracted from video could provide additional information as video consists of more frames compared to image, whereby the motion patterns can only be exploited from these successive number of frames. In light of fact, majority of hand-crafted procedures are actually unable to extract high-level distinctive information from raw frames due to various problems like high complexity and noises. While deep learning has significantly grown with the digital era that has brought about big data in all forms, it has attained great attentions among researchers for solving various video analysis tasks.

Deep networks such as Convolutional Neural Networks (CNN) have found to be one of the outstanding methods for feature extraction in such a large and complex data. They have yielded impressive results in performing action classification on videos. Several deep models are able to outperform the performance achieved by iDT. For instance, Convolutional 3D (C3D) (Tran et al. 2015), two-stream Inflated 3D ConvNet (I3D) (Carreira & Zisserman 2017) and R(2+1)D. Each network architecture will be discussed in detail in the following.

### C3D Network Architecture.

Prior to the invention of 3D convolutions, several works are proposed to solve video classification tasks by using 2D convolutions. This is due to the reason that 2D convolutions are proven to be an effective approach to extract features from images. Therefore, researchers attempted to reimplement the 2D convolutions into the video classification domain. For example, (Karpathy et al. 2014) proposed to perform feature extraction on the rescaled frames of the video independently using 2D CNNs. However, the performance of 2D-based approaches became irrelevant and obsolete rapidly as it failed to capture the complex motion information from the video inputs. (Tran et al. 2015) solved this issue by proposing a novel architecture known as Convolutional 3D (C3D). In this work, deep 3D convolutions that are efficient in capturing spatiotemporal information, are proposed as the main framework for complex feature learning.

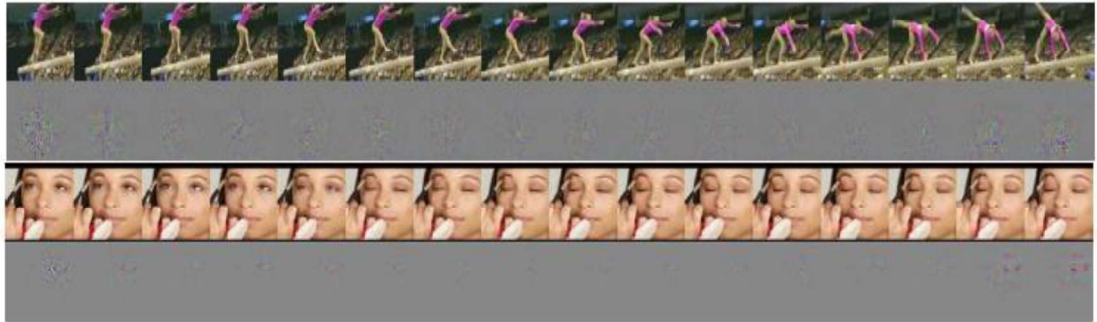
3D convolutions, also known as 3D ConvNet, are able to accurately handle automated recognition of human actions. This could be achieved by repetitively convolving 3D kernels over stacked adjacent frames in order to extract spatial and temporal information from the streams. In other words, the feature maps generated in each layer are connected to the one in the previous layer, thereby capturing the complex motion information. Similar to 2D convolutions, each 3D kernel can only extract a single feature since the kernel weights are identical across the entire stack of frames. Thus, multiple 3D convolutions with different kernels are required to be implemented in order to extract distinct features across the frames.

Owing to the outstanding achievement of 3D ConvNets in modelling spatiotemporal information, (Tran et al. 2015) further extended this knowledge by implementing it as C3D. In this architecture, homogeneous convolutional kernels with a size of  $3 \times 3 \times 3$  and 3D pooling of size  $2 \times 2 \times 2$  are applied at each layer. Such settings are believed to be the best option for spatiotemporal feature extractions after several experiments. As such, the temporal information in the early phase could be preserved. The network architecture is as shown below.



**Figure 2.5 C3D architecture (Tran et al. 2015)**

One of the interesting ideas in this architecture is the way how it retains the temporal information across the contiguous frames. Using deconvolution method, the visualization of the internal learning process of C3D is as shown in the figure below.



**Figure 2.6 Visualization of C3D, using the method of deconvolution  
(Tran et al. 2015)**

It could be clearly seen that C3D filters selectively focus on both appearance and motion by concentrating the appearance in the first place, and then tracing the noticeable motion in the next frames. Apparently, it differs from standard 2D ConvNet which focuses only spatial information. Thus, it is great in representing distinct video features for various video-related tasks such as event detection and action localization.

In the domain of action recognition, the features of C3D are extracted and evaluated on UCF101 dataset which consist of around 13k videos of 101 human actions classes. A simple multi-class linear SVM is used for training the model. Owing to the efficiency of C3D in capturing high-level semantic information from videos, it has outperformed iDT and achieved the state-of-the-arts.

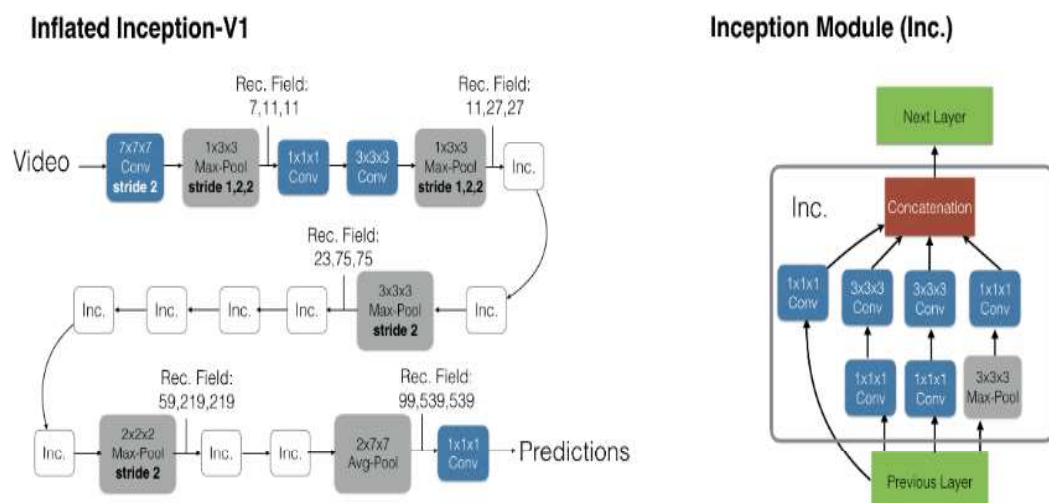
### **I3D Network Architecture.**

Over the years, training on top of a pretrained model has been consistently shown to significantly outperform models that are trained from scratch. Other than reducing the training time for a model, it also performs better on specific problems, thanks to the great achievement of the pretrained model. While image representation architectures have grown rapidly, there is still no clear front-runner for video classification tasks back in the days. Thenceforth, there is no top-performing pretrained 3D-model for solving video recognition tasks at that particular moment.

Instead of doing numerous painstaking trial and error attempts on building a 3-D model for video classification, I3D (Carreira & Zisserman 2017) performs the inflation of 2D filters that are trained from the ImageNet, which then allows it to make use of a pretrained 2D model for building a 3-D model. This could be accomplished by expanding all the filters and pooling kernels from 2D ConvNets into 3D ConvNets by means of adding an additional temporal dimension. To illustrate, the 2D filters with a dimension of  $N \times N$  are just converted into 3D filters with a dimension of  $N \times N \times N$ .

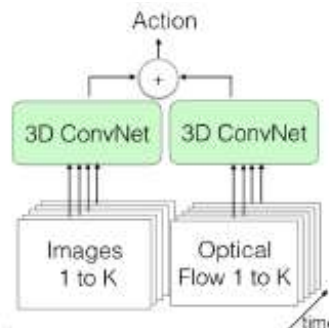
Since many of the deep image classification models like VGG-16, Inception and ResNet has gained notable achievements while using ImageNet pretrained model as their subcomponents, (Carreira & Zisserman 2017) also attempted to use the pretrained model as the backbone architecture of I3D. With transfer learning, the weights and biases of an Inception-v1 model which has been implicitly pretrained on ImageNet, is loaded for better initialization. The novel architecture is believed to be an extremely deep network due to the nature of Inception-v1 network architecture. The overall architecture of Inflated Inception-v1 is as shown below.

A sequence of video is generated by a repetitive duplication of single image input. This could be achieved by iterating the 2D filters weights  $N$  times along the dimension of time, and rescaling them by a division of  $N$ . As such, the overall network response could be identical with the one on the original single image input.



**Figure 2.7 Inflated Inception-v1 on the left and its detailed inception submodule on the right (Carreira & Zisserman 2017)**

Besides, in order to obtain more sense of recurrence and more accurate results, two 3D streams are used, whereby one is trained on RGB inputs, and another is on the optical flow inputs. The predictions from both networks are averaged in the end.



**Figure 2.8 Two-Stream I3D architecture (Carreira & Zisserman 2017)**

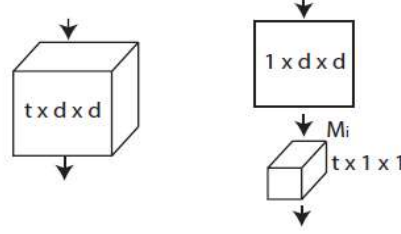
In actual fact, I3D currently holds the best results in action recognition. Since two-stream I3D has acquired the best performance among the state-of-the-art, this project will be implemented based on the two-stream network configuration for obtaining better results. However, the replacement of the network architecture will be made owing to the nature of this project which requires a real-time performance, and the details of the architecture will be discussed later.

### **R(2+1)D Network Architecture.**

Although CNNs have been widely used for feature construction due to its superior performance on visual objects, 2D CNNs are incapable of capturing temporal and motion information. Unfortunately, temporal and motion information are considered as primary components of video analysis. By performing 3D CNNs over the spatiotemporal video volume, the features of both spatial and temporal dimensions could be acquired. However, there are some primary downsides that should be reflected for this network. In contrast to 2D CNNs, 3D CNNs require a higher computational cost and more excessive memory consumption during the training process. In other words, it necessitates a greater number of parameters during the training process, which results in an increased complexity of the optimization process.

To overcome the aforementioned problems, (Tran et al. 2018) proposed a novel architecture, which is R(2+1)D, whereby it factorizes 3D convolutional filters into two separate operations which are a 2D and a 1D convolution. Full 3D convolutions are likely to be more convenient and cheaper by breaking down 3D convolutions into

separated tasks. They trained the model proposed on both RGB and OF inputs using two-stream framework which would be discussed later, then combined the prediction scores generated by each stream through average fusion.



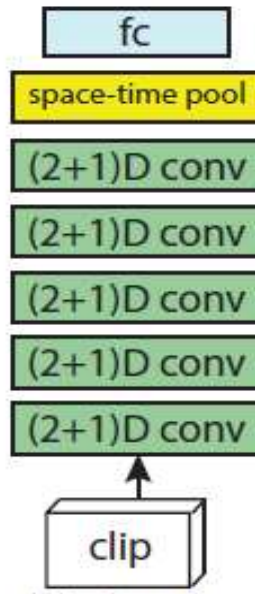
**Figure 2.9 (2+1)D convolution (Tran et al. 2018)**

Instead of using a full 3D convolutional filter with a size of  $N_{i-1} \times t \times d \times d$ , they substituted it by using a  $M_i$  2D spatial convolutional of size  $N_{i-1} \times 1 \times d \times d$  and  $N_i$  1D convolutional filter of size  $M_i \times t \times 1 \times 1$ , given by:

$$M_i = \left\lfloor \frac{td^2 N_{i-1} N_i}{d^2 N_{i-1} + t N_i} \right\rfloor \quad (1)$$

whereby  $N_i$  represents the filters number applied on  $i$ -th convolution block,  $t$  depicts temporal extent and  $d$  represents the height and width. In such circumstances, the formula of  $M_i$  is made to ensure the number of parameters used in (2+1)D block is closely equivalent to the one in full 3D convolution block as well as to obtain a compact-sized model in the end. Besides, if any striding is involved in the 3D convolution, the striding will be correspondingly decomposed into its spatial or temporal dimensions as well.

In the work, homogeneous spatiotemporal residual blocks are used where they eliminated bottlenecks such as vanishing gradients. Each block consists of 2 convolutional layers and the ReLU activation function is applied after every convolutional layer in the block. After convolving over multiple convolutional blocks, a global average pooling is applied on the final output of last convolutional layer in order to generate a fixed-size representation to feed the FC layer for final classification output.

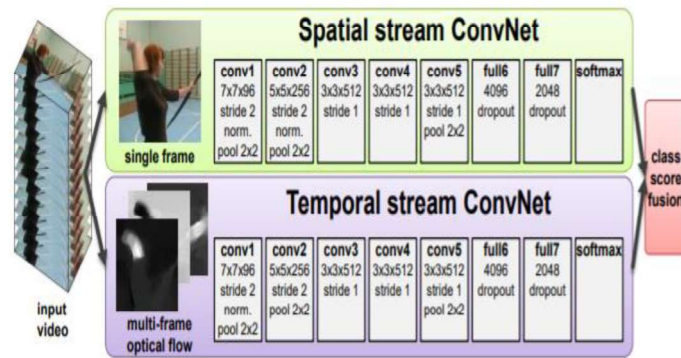


**Figure 2.10 R(2+1)D which are ResNets with (2+1)D Convolutions**

Owing to the fact that the additional ReLU activation function is applied at each 1D and 2D layers, R(2+1)D is capable of representing more complex functions as it offers an additional non-linear rectification without modifying the number of parameters used in 3D convolutional block. Furthermore, it yields a lower training and testing error compared to a full 3D convolution. Thus, the optimization on (2+1)D blocks could be done easier compared to full 3D blocks. In short, the proposed architecture, R(2+1)D achieves a superior result compared to the state-of-art performance in human action recognition.

### **Modality and Temporal Fusion.**

Video can be instinctively broken down into two components, which are the spatial and temporal components. The spatial part carries information about the relationship between scenes and objects portrayed in a video in the form of static and individual frames. On the other hand, the temporal part conveys the movement and interactions between multiple objects and its environment in the form of motion across multiple frames. Thus, (Simonyan & Zisserman 2014) proposed a two-stream architecture to divide video classification architecture into two streams such as Spatial Stream and Temporal Stream, whereby each of the streams is being implemented using a deep ConvNet and the class-scores are combined using average fusion or SVM.



**Figure 2.11 Two-stream architecture in video classification**  
(Simonyan & Zisserman 2014)

For Spatial Stream ConvNet, it processes the static raw frames of the video individually to perform action recognition from 2D images as they believed that most actions are strongly associated with certain objects. On the other hand, Temporal Stream ConvNet mainly focuses on the motions between multiple video frames, whereby it receives stacked optical flow-based frames as inputs and extracts motion information as its output. Based on the paper, it could be deduced that training a temporal ConvNet using optical flow-based frames yields a better accuracy than training on raw stacked frames.

### 2.3 Object Detection

In previous years, there has been a speedy and tremendous expansion in the domain of object detection due to the advancement of deep learning. Unlike image classification or object localization which only predicts the class of an object or locates the presence of objects in an image, object detection combines both of the tasks mentioned earlier. In other words, it comprises of localizing and classifying one or more objects presented in an image. Various innovative deep learning researches and algorithms have been established for implementing efficient object detection.

In this project, object detection will be implemented as a real-time object counter for monitoring the number of occupants and seats in a conference room from time to time. Deep insights of meeting room usage could be gained through the information collected. A number of recent top-performing deep-learning-based object detection techniques will be presented in this section including RCNN, Fast-RCNN, Faster-RCNN and YOLO.

**RCNN-Family.**

Due to the success of CNNs in the tasks of image classification, researchers have started to employ CNN for object detection to a greater extent. Region-Based Convolutional Neural Network (RCNN), which was proposed by (Girshick et al. 2014), is one of the state-of-the-art object detection approaches that is built based on CNNs. RCNN algorithm requires the extraction of region proposals which are bounding boxes that potentially contain objects via Selective Search. Each region is then being reshaped as the input size for CNN before passing to the network and the features extracted for each region could be classified using SVM. Meanwhile, a bounding-box regression algorithm will be applied to refine the location of bounding boxes for each identified region. However, it is highly computational expensive and time-consuming as it has to extract features in each candidate region on every single image.

Fast-RCNN (Girshick et al. 2015) solved this by integrating ROI Pooling layer at the end of deep CNN for feature extractions on the generated activation map. However, the inference time for Fast-RCNN is still dominated by the time to generate the region proposals. In addition, it is still made up of multi-stages, thus leading to slow training process.

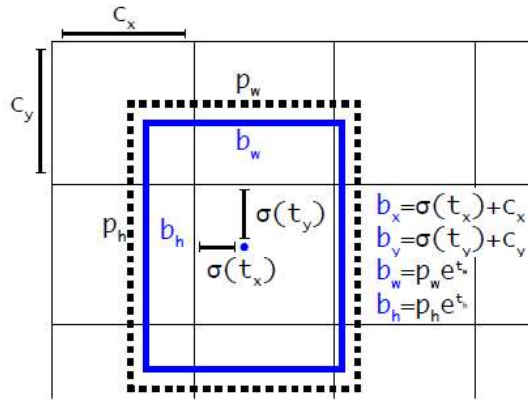
Faster-RCNN (Ren et al. 2016) truly performs end-to-end training where a separate network namely Region Proposal Network, or RPN, is designed for extracting the object proposals, along with an objectness score for each proposal. In this network, the idea of K anchor boxes has been introduced for dealing with the variations in scale and aspect ratio of objects. However, it still falls short of a real-time detection performance due to huge time consumption for generating different object proposals.

**YOLOv3.**

Up to the present, all the methods discussed above are made up of two stages where a set of object proposals will be generated first before sending to classification or regression heads. These methods are time-consuming and difficult to be optimized as multiple models have to be trained separately before making predictions. Thus, a single-staged model, namely You Live Only Once (YOLO) which was proposed by (Redmon et al. 2016), had unified some benefits over the traditional ways of object recognition. YOLOv3 will be selected as the main architecture for object detection in this project due to its rapid detection speed and decent accuracy. In fact, YOLOv3 deals with object

detection in a different way compared to RCNN family. It is an object detector that uses feature learned from deep CNN to detect and classify objects.

Instead of typical region proposal method, YOLOv3 takes the entire image as an input and makes predictions on the class probabilities and coordinates of each bounding boxes. This is achieved by the concept of dividing the image into  $S \times S$  grid cell. Each particular grid cell will be responsible for detecting objects, depending on the location of the object center. Anchor boxes are being applied in this architecture via K-Means clustering. Later, several bounding boxes from each grid cell would be retrieved from the anchor boxes generated. The network will then perform predictions on each bounding box to acquire its objectness score and offset values such as  $b_x$ ,  $b_y$ ,  $b_w$  and  $b_h$ , where the equations for each offset are as shown in the diagram below.



**Figure 2.12 Bounding boxes with location prediction and dimension priors**  
(Redmon et al. 2016)

Based on the diagram,  $b_x$ ,  $b_y$ ,  $b_w$ ,  $b_h$  represents the x, y center coordinates, width and height of the prediction in a bounding box respectively. The center coordinates are predicted using a sigmoid function. Then, the network outputs are denoted as  $t_x$ ,  $t_y$ ,  $t_w$  and  $t_h$  respectively, whereas  $p_w$ ,  $p_h$  are anchors dimensions for the box.

In fact, YOLOv3 makes predictions at 3 different scales, which have been generated by downsampling on the input image with strides of 32, 16 and 8 respectively. A  $1 \times 1$  kernel is applied on 3 different sized feature maps in order to acquire the final outputs for detection. Hence, predictions will be made on a total of 9 anchor boxes (3 scales and 3 anchors). The eventual output of the network would be a feature map which has similar size as the previous feature map. It contains  $(B \times (5 + C))$  entries, where  $B$

denotes the number of bounding boxes for predictions, 5 represents the objectness score and 4 bounding box offset values, and C is the number of classes for detections.

In addition, by replacing the final classification layer with independent logistic classifiers such as logistic regression, YOLOv3 is able to perform multilabel classification in a more efficient manner. Threshold is also applied for filtering out the bounding boxes with lower confidence score. As such, it is able to handle more complex data.

Other than that, a more powerful network, DarkNet-53, is introduced as its feature extractor in order to improve the detection accuracy. The network architecture is as shown below.

	Type	Filters	Size	Output
	Convolutional	32	$3 \times 3$	$256 \times 256$
	Convolutional	64	$3 \times 3 / 2$	$128 \times 128$
1x	Convolutional	32	$1 \times 1$	$128 \times 128$
	Convolutional	64	$3 \times 3$	
	Residual			
	Convolutional	128	$3 \times 3 / 2$	
2x	Convolutional	64	$1 \times 1$	$64 \times 64$
	Convolutional	128	$3 \times 3$	
	Residual			
	Convolutional	256	$3 \times 3 / 2$	
8x	Convolutional	128	$1 \times 1$	$32 \times 32$
	Convolutional	256	$3 \times 3$	
	Residual			
	Convolutional	512	$3 \times 3 / 2$	
8x	Convolutional	256	$1 \times 1$	$16 \times 16$
	Convolutional	512	$3 \times 3$	
	Residual			
	Convolutional	1024	$3 \times 3 / 2$	
4x	Convolutional	512	$1 \times 1$	$8 \times 8$
	Convolutional	1024	$3 \times 3$	
	Residual			
	Avgpool		Global	
	Connected		1000	
	Softmax			

**Figure 2.13 Darknet-53 (Redmon et al. 2016)**

In fact, there is a total of 53 convolutional layers in the network architecture. Besides, batch normalization and shortcut connections are applied. The output from this extractor would be a tensor encoded with the coordinates of bounding box, class prediction score and objectness score. The new feature extractor is more efficient than ResNets as it is almost 2x faster than ResNets while achieving the similar proficiency in the domain of object detection. As a conclusion, YOLOv3 is able to outperform the two-staged object detection models due to its rapid detection speed and decent detection accuracy.

## CHAPTER 3 MEETING ROOM EVENT DETECTION

### 3.1 System Overview

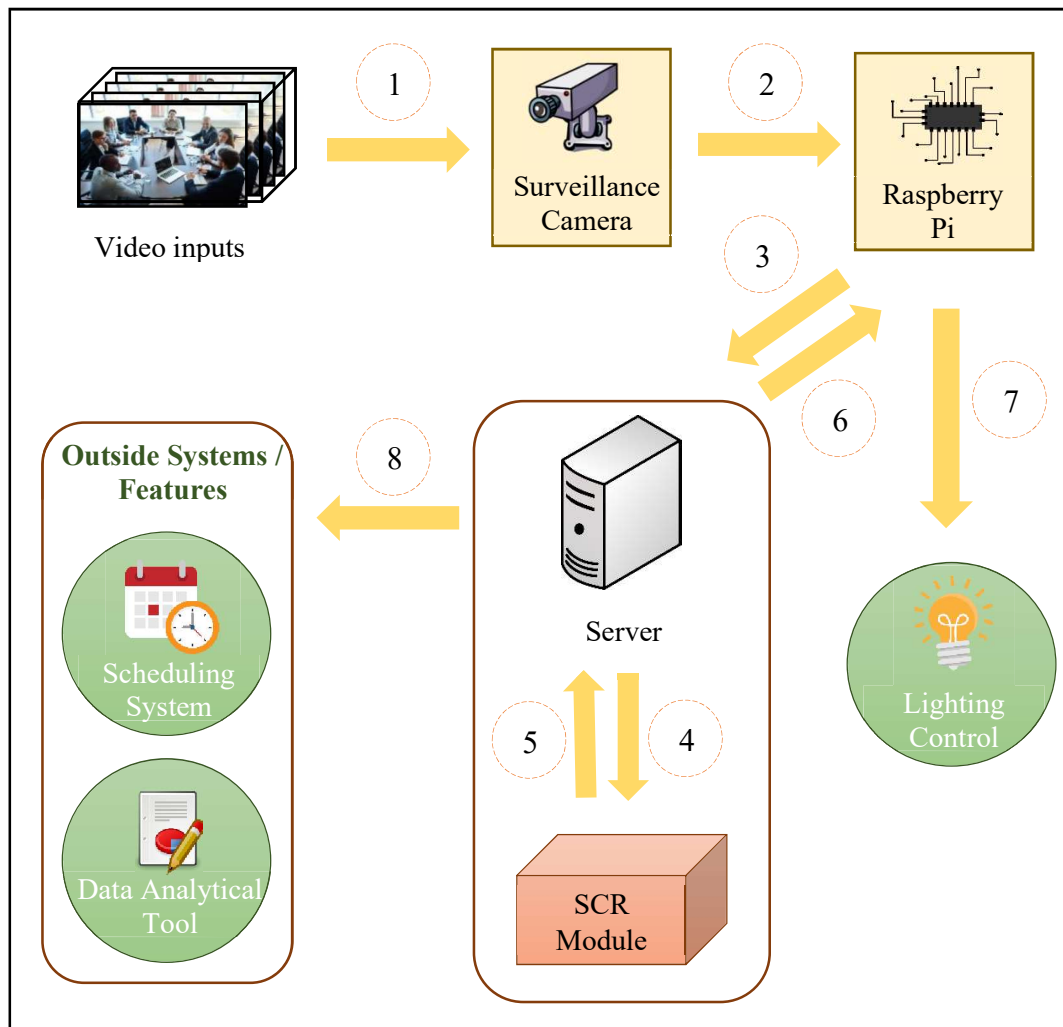


Figure 3.1 Overview of system

A mountable surveillance camera (Raspberry Pi camera) could be placed in several strategic points of ceiling or walls throughout the meeting rooms. Several cameras are ideal to be placed independently on different location, in order to record videos from different angles in the wide area. Connected to a Raspberry Pi 3B+, the footages are transferred and stored in a database server which are then being fed to the Smart Conference Room (SCR) Module and the details of the module will be discussed later in Section 3.2.

In this module, deep models which are built on CNNs, are trained to detect and classify human actions through video analysis. Object recognition will be implemented in order to calculate the number of seats and also the number of occupants presented in the meeting room. After identifying employees' actions in the meeting rooms, the module will then generate a set of results to state whether the meeting rooms are being utilized properly or not. For example, if an employee is discovered for not using the room appropriately, the module will then initiate signals to the backend server. This triggers the office scheduling system to immediately update and make the room available again in the system in order for other employees to proceed with ad-hoc bookings. Besides, the electricity in the room will be cut off automatically using Raspberry Pi 3B+, not only for raising awareness of the abuser but also for conserving and sustaining company resources.

Apart from that, the system may be integrated with data analytical tool for generating useful analytics. This could be achieved by utilizing the information given by both event and object detections. All of these could be spontaneously documented in weekly and monthly reports by the other existing system in order for companies to gain insights into the underlying patterns of meeting room utilization. The reports may display who is using the room, what is his/her activities in the room and when did he/she use the room. With these analytics, companies can make informed decisions based on meeting room usage in order to optimize space better, improve team collaborations and enhance productivity.

### 3.2 Event Detection for SCR

In this section, Section 3.2.1 will describe the targeted events to be detected in this system. Next, Section 3.2.2 will propose a network architecture for handling event detections in SCR.

#### 3.2.1 Targeted Events

Recently, most of the existing SCR systems are restricted to only detect the presence of occupants instead of their activities, thus resulting in the wastage of company resources and loss of productivity. In order to ensure proper usage of meeting rooms, a smart conference room is developed in this project using the technique of human action recognition. As such, the on-going events in conference rooms could be detected and recognized via the motion information collected from several surveillance cameras.

Besides, useful information could be generated for gaining in-depth analysis about the room usage. It may be further integrated with other system for additional functionalities. For instance, lighting control system can react differently based on the event detected. If meeting activities are detected, the lighting control system can adjust and increase the brightness of the lighting, not only allowing presenters to conduct presentation with their materials prepared, but also allowing others to see clearly. On the other hand, if the action class detected is non-meeting or idle, the lighting system and electricity power may be shut down in order to preserve the resource energy. In fact, this system can also be integrated with reservation scheduling system in order to keep track of the availability of the rooms based on the on-going events detected.

As mentioned earlier, a large-scaled dataset which consists of multiple footages in conference rooms is required for training the model. However, the footages of conferencing activities in an organization are usually private and not publicly accessible. Hence, the dataset is needed to be self-generated from scratch via extensive data collection, data cleaning and data annotation. In this project, data collection is done in collaboration with Company X. Approximately, there is a total of 200 hours footages from 6 different meeting rooms are collected from Company X. Later, the footages are processed into proper dataset by exhaustive data cleaning and data annotation. The entire process is time-consuming as the data collected is needed to be gone through one

by one and labeled manually. This is due to the reason that a single video clip may contain multiple instances within an arbitrary time frame.

There are 3 event classes to be detected and classified in this project. To illustrate,

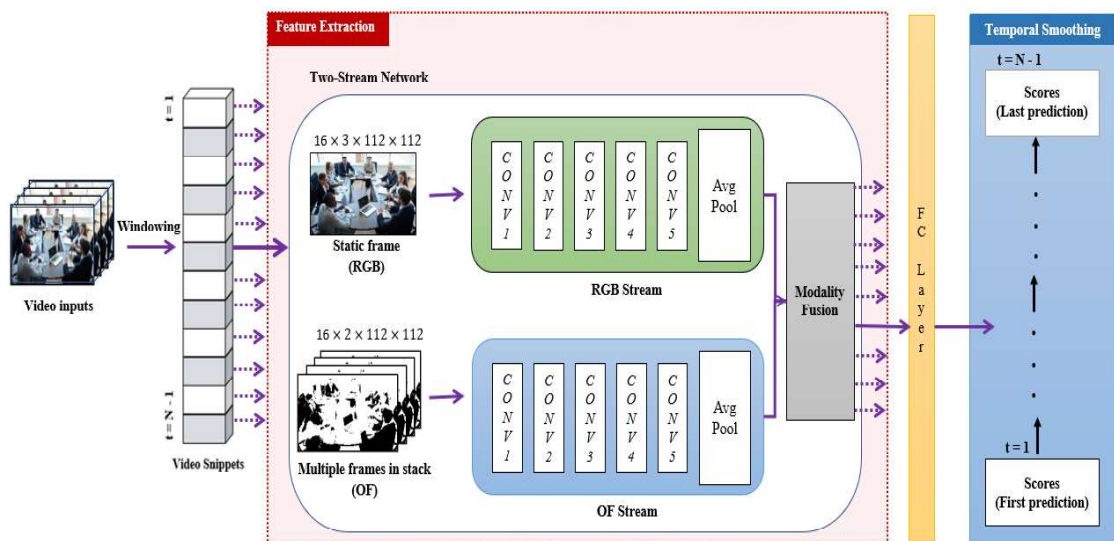
- Meeting: Consists of activities such as stand-up meetings, all-seated meetings, presentations and using laptops.
- Non-meeting: Consists of activities such as ‘squatting’ or monopolizing the room for privacy usage, room cleaning and equipment maintenance.
- Empty: In the case where there is no any occupant in the meeting room.

Initially, this project is planned to handle passive activities like ‘room cleaning’ and ‘equipment maintenance’. However, since the set of footages provided by company X has insufficient instances of such activities, the system is unable to detect for both of the classes up to this moment. Thus, these events are considered as non-meeting activities for now.

Note: The name of company X is made anonymous at the request of the company itself.

### 3.2.2 Proposed Event Detection Method

In this work, the proposed architecture for event detection is two-stream R(2+1)D network architecture, which is the factorization of 3D convolutional layers into a block of 2D and 1D convolutions in both RGB and OF stream. The network architecture is described as shown below.



**Figure 3.2 Overview of proposed model in SCR module**

**Video Snippets Extraction (Windowing).** Video frames are extracted from the sequences of video streams to form multiple short video snippets, whereby each of the snippets contain  $L = 16$  frames. Each of the video snippets are converted to RGB and optical flow (OF) inputs for different modalities. Farneback's algorithm is applied on the frames for dense OF computations. As such, RGB stream is responsible for acquiring information about appearance presented in videos whereas OF stream captures the interaction among them by comparing the displacement difference between the scene and objects. The frames will be rescaled into size of  $128 \times 171$ , then be further resized into  $112 \times 112$  by random cropping. Eventually, the size of RGB frames are  $16 \times 3 \times 112 \times 112$  while the size of OF frames are  $16 \times 2 \times 112 \times 112$  as the OF frames will be gray-scaled. Figure below shows the sample outputs for RGB and OF (horizontal and vertical) frame.



**Figure 3.3 RGB, OF-Horizontal, OF-Vertical frames of "Meeting".**

**Feature Extraction.** In SCR Module, the two-stream R(2+1)D-34 network architecture is implemented in order to capture motion in forms of spatial and temporal. In this case, the inputs of each stream would be multiple clips that consist of 16 RGB and OF frames respectively in each clip. Each of the stream then produce a feature vector by convolving over multiple blocks of (2+1)D convolutions.

As mentioned earlier, (2+1)D convolution is achieved by decomposing 3D convolutional operation into 2 successive steps, which are 2D convolution and followed by 1D convolution. In this network architecture, the input clips have the size of  $3 \times L \times H \times W$  whereby 3 denotes the RGB channels,  $L$  is the frame counts,  $W$  and  $H$  stands for the width and height respectively. In this project, the network architecture is configured using ResNet-34 instead of ResNet-18 and ResNet-152, owing to the consideration of its efficiency in terms of the real-time implementation and detection accuracy. The network configuration is shown in Figure 3.4 and Figure 3.5

LAYER NAME	OUTPUT SIZE	R(2+1)D-34
conv1	$L \times 56 \times 56$	$3 \times 7 \times 7, 64, \text{stride } 1 \times 2 \times 2$
conv2_x	$L \times 56 \times 56$	$\begin{bmatrix} 3 \times 3 \times 3, 64 \\ 3 \times 3 \times 3, 64 \end{bmatrix} \times 3$
conv3_x	$\frac{L}{2} \times 28 \times 28$	$\begin{bmatrix} 3 \times 3 \times 3, 128 \\ 3 \times 3 \times 3, 128 \end{bmatrix} \times 4$
conv4_x	$\frac{L}{4} \times 14 \times 14$	$\begin{bmatrix} 3 \times 3 \times 3, 256 \\ 3 \times 3 \times 3, 256 \end{bmatrix} \times 6$
conv5_x	$\frac{L}{8} \times 7 \times 7$	$\begin{bmatrix} 3 \times 3 \times 3, 512 \\ 3 \times 3 \times 3, 512 \end{bmatrix} \times 3$
	$1 \times 1 \times 1$	spatiotemporal pooling, FC layer with softmax

Figure 3.4 R(2+1)D-34 architecture (Tran et al. 2018)

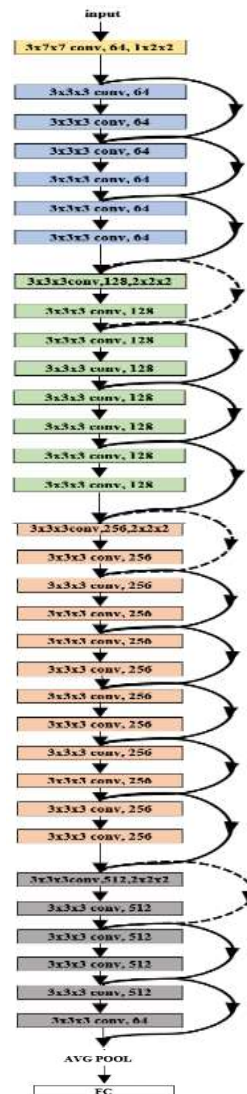


Figure 3.5 R(2+1)D with 34 layers specifications

In addition, downsampling is required for reducing the size of inputs, which then results in shorter time and lesser memory consumption while training. The implementation of

R(2+1)D is actually similar to the one in R3D. However, if any striding is involved in R3D, the striding will be correspondingly decomposed into its spatial or temporal dimensions, allowing (2+1)D convolutions to have the similar resulting dimensions. Instead of applying down sampling at every layer of this network, it has been applied only on the spatial layer at the first convolutional layer with convolutional striding of  $1 \times 2 \times 2$  and on the spatial layer in third, fourth and fifth block of the architecture with convolutional striding of  $2 \times 2 \times 2$  respectively. The last output produced by this network architecture would be a convolutional tensor with size of  $\frac{L}{8} \times 7 \times 7$ . After convolving over multiple convolutional blocks, average pooling is applied to yield a 512-dimensional feature vector. Later, modality fusion would be performed in order to fuse the feature vectors from both streams via averaging. The fused feature vector generated for each and every time step will then be fed into FC layer for final action class prediction.

**Temporal Smoothing.** In order to smoothen out the prediction score, the Exponentially Weighted Moving Average (EWMA) is applied. It could be achieved by weighing the number of prediction scores and averaging them. Generally, it is effective for time-series data. It holds the past values in memory buffer and constantly updates the buffer whenever a new prediction is obtained. The equation of EWMA is shown below:

$$S_t = ax_t + (1 - a) S_{t-1} \quad (2)$$

In this equation,  $S_t$  depicts the value of moving average at time  $t$ . It could be obtained by multiplying the previous value of moving averaging and the value of raw signal  $x_t$  at the specific time step. The power of “smoothing” is controlled by the parameter  $a$ . To illustrate, if  $a$  is an extremely small value, the smoothing effect would be very strong, and vice versa. Thus, the abrupt changes in prediction scores could be avoided by implementing this algorithm.

### 3.3 Dataset Generation and Preparation

**Data Collection.** Collected in Company X, the conference dataset consists of at least 200 hours of Raspberry Pi camera footages from diverse viewpoints in 6 different meeting rooms. For example, the camera is directly placed in front of a meeting table in order to capture the movement of participants clearly. Besides, some of cameras are

positioned at a range of around 45 to 75 degrees from a tall-sized cupboard in order to obtain an unobstructed viewing area. The footages are with size of  $640 \times 480$  and each of them is approximately 3 to 6 hours long. Several events could be acquired in each video including Meeting, Non-Meeting and Empty. However, extensive data cleaning and annotation have to be conducted in order to generate a valuable yet meaningful dataset. Subsequently, the generated dataset is split into training, validation and testing sets with a percentage of 60%, 28% and 12% respectively.

**Data Cleaning and Annotation.** Since some of the footages collected in Company X contain noises such as obstructive or corrupted videos, data cleaning should be carried out to eliminate the flaws. There is a total of 60GB camera footages and each of them were being gone through one by one in detail in order to attain the useful instances. After looking into the datasets successively, it is discovered that the datasets comprise of only less variations of activities to be classified, which is an unexpected circumstance. Initially, it is presumed that the events could be detected in a finer manner. For instance, room cleaning and equipment maintenance. Nonetheless, there are only a handful of samples discovered for such events. Apparently, it is insufficient for training the model. Hence, the final decision made is just to select several labels which are coarser, such as ‘Meeting’, ‘Non-Meeting’ and ‘Empty’.

During data annotation, the selected footages with useful information are recorded in a csv file, along with its video\_id, start\_time, end\_time and label. Despite the fact that the tasks were fairly distributed among the team, the entire process on annotating these footages still costed about 5 weeks to be accomplished. In the end, the dataset generated consists of 2851 instances, whereby each of them is at least a 6-second clip. There are 1146, 1128 and 577 samples for Empty, Meeting and Non-Meeting classes respectively.

**Data Preprocessing.** After data annotation, the useful footages are then being extracted using a written python script and FFMPEG. Each of the clips are resized to  $320 \times 256$ . Subsequently, the frame rate of each videos is converted to 25 frame per second (FPS). In order to avoid bottlenecks, the dataset is being preprocessed before training.

During data preprocessing, video frames are extracted from the entire video and saved to a sequence of RGB and OF images. The OF frames are computed using Farneback’s algorithm. After preprocessing, the data will then be used to finetune the pretrained model in order to extract more features and classify a number of action classes.

### 3.4 Experimental setup

**Training setup.** The Kinetics pretrained model is finetuned with Conference Dataset collected from Company X. The learned weights and biases of the pretrained model will be completely transferred to the current model via transfer learning. Besides, batch normalization is implemented on every layer. The size of minibatch is set to 4 clips owing to the limited computing resources. Stochastic Gradient Descent (SGD) optimizer is being used and the finetuning process is set to be completed in 45 epochs with the initial learning rate of 0.01. The time taken for entire training process is presented in the diagram below.

Modality	Duration (hours)
RGB stream	18
OF stream	18.5

**Table 3.1 Time taken for training process**

**Testing setup.** During the testing process, in order to get more accurate results, the output scores from both RGB and optical flow streams are fused together through averaging. This experiment only reports top-1 clip accuracy and top-1 video accuracy, ignoring the conventional top-5 accuracies in video analysis task as there are only 3 classes to be predicted. In order to get the video top-1 accuracies, center crops of 10 clips are uniformly sampled from the video and the 10-clip predictions are averaged to obtain the final predictions. Besides, 10-crops testing method which obtains a center and 4 corner crops from the clip and the other 5 crops from the horizontal reflections of the clip, is applied at the final stage of experiments.

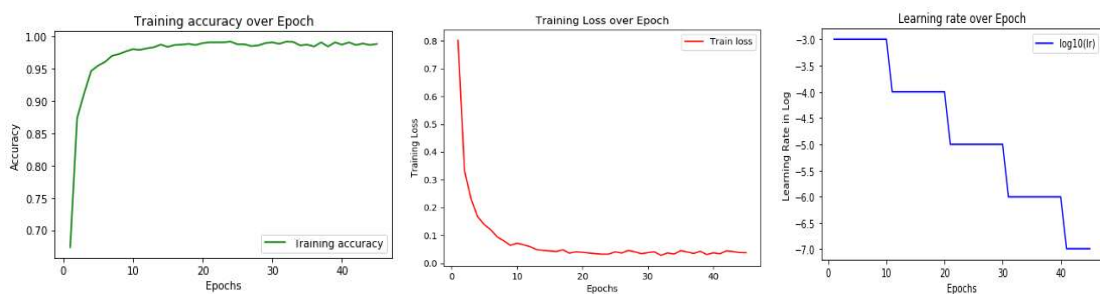
### 3.5 Experiments

This section discusses about various configurations on both RGB and OF stream in order to acquire the best performance for the Conference Dataset. The details for each stream are provided in Section 3.5.1 and 3.5.2 respectively. Section 3.5.3 describes the fusion of both streams.

#### 3.5.1 RGB Network Configuration(s)

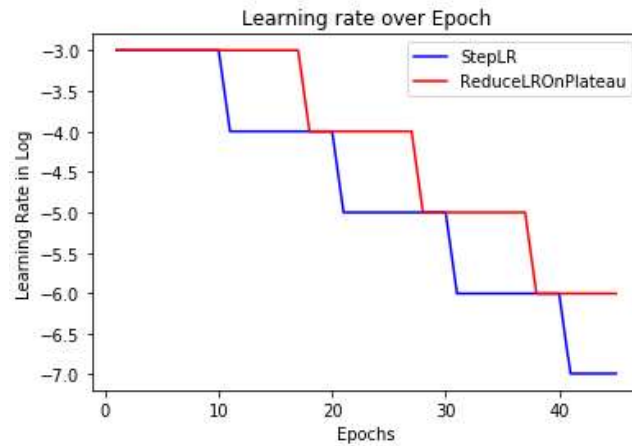
Several experiments for RGB network are conducted using different settings in the clip-length ( $L$ ), freezing layers in pretrained model, dropout value ( $p$ ) and types of scheduler.

First of all, the learned weight and biases of Kinetics pretrained model are transferred to the current model. It is done without freezing any layers in the first place so that the network is able to update its parameters freely. In the experiments, the base learning rate of the training process is set to 0.01. The entire finetuning process is completed in 45 epochs. Initially, the network is configured with ( $L = 8$ ), without any freezing layers and dropout applied. StepLR is applied to reduce the learning rate by a factor of 10 per every 10 epochs. The training process is validated by visualizing the graphs of training accuracy and training loss over epochs, so that it could be proceeded with other settings for improvements. The stepLR scheduler is also visualized below.



**Figure 3.6 Training accuracy, training loss and stepLR scheduler**

The clip-length is tested with either  $L = 8$  or  $L = 16$  as they are the only feasible values for real-time detection speed. This is probably due to the reason that the Raspberry Pi used to record footages in meeting rooms is set to be 25 FPS. Thus, in order to acquire the highest detection speed, the clip-length selected must be lesser than 25. Meanwhile, the freezing point of the model is set at *conv\_2* to investigate if the performance will improve if part of the weights from the pretrained models are retained. Instead of a plain StepLR scheduler, a smarter approach, ReduceLROnPlateau, is applied whereby the scheduler will only reduce the learning rate when the optimizer is unable to improve. The scheduler is set to wait for 10 epochs (patience). The learning rate will be reduced by a factor of 10 if the model does not improve for 10 epochs. It is visualized in Figure 3.7.



**Figure 3.7 Comparison between two LR schedulers**

Based on the graph, it shows that 10 epochs are insufficient to warm up the model to train well. In addition, several data augmentations such as random horizontal flipping, random brightness augmentation and random color shifting on RGB channels, are applied as the model with only random cropping and temporal jittering are unable to generalize well since there are too few variations in the Conference Dataset. However, only random horizontal flipping could slightly improve the clip accuracy. In order to reduce the capacity of the model from overfitting the training data, a dropout is being implemented with values of  $p = 0.5$  and  $p = 0.9$ . After several experiments for tweaking the performance of this network, the top-1 clip accuracy and video accuracy for each configuration are reported as below.

Freezing Point	Clip-Length ( $L$ )	Dropout	Clip Acc.	Video Acc.
None	8	-	69.5%	71.3%
None	16	-	74.9%	76.6%
Conv2_x	16	-	78.6%	82.4%
Conv2_x	16	0.5	83.8%	89.7%
Conv2_x	16	0.9	82.3%	87.4%

**Table 3.2 Results of RGB network with different settings**

Based on the results shown above, it could be deduced that the RGB model with best performance is configured to retain the learned weights and biases at *conv\_2*, with settings of  $L = 16$  and  $p = 0.5$ .

### 3.5.2 OF Network Configuration(s)

In this work, the OF network is experimented using different settings for clip-length ( $L$ ) and dropout values ( $p$ ). Initially, this network is configured with a clip-length of  $L = 16$  as it is proven to be better while training the RGB network. The Kinetics pretrained model without freezing any layers is used by applying transfer learning. Initially, the learning rate is adjusted to be 0.01 and ReduceLROnPlateau is applied with a patience of 10. The whole training process is configured to be 45 epochs. However, it is discovered that the model is unable to converge, which is a situation that never occurred while training on the UCF101 Dataset. Hence, in-depth examinations are carried out in order to figure out the problems. Eventually, it is found that the low performance of OF computations is mainly dominated by the dataset generated, as the raw footages are collected with at least 100 FPS using Raspberry Pi. Thus, re-generation of the dataset with a sampling rate of 5 is performed in order to resolve the issue. Later, the training process is able to be achieved on the latest version of dataset without any problems.

Similarly, different clip-length settings such as  $L = 8$  and  $L = 16$  are configured to test if the OF network has the same outcome as RGB stream. It turns out to be similar with the one in RGB stream as expected, due to its sensitivity towards the extent in capturing the motion information. The freezing point is set to be *conv\_2* to test if it could perform better. In addition, several regularization techniques are applied such as dropout, random horizontal flipping, random brightness augmentation and random color shifting. Nonetheless, all of these techniques are unable to improve the model performance since the color and brightness shifting would not influence the OF computations. In order for the model to exploit the rectification non-linearities better, the inputs of the network should be zero-centered. In the case of optical flow, the motion vectors are already zero centered as the positive and negative values in the give motion vectors are equally distributed by nature. However, global motions are considered to be more complicated because it will cause the motion vectors to be dominated by a particular movement such as camera motion. There are various techniques can be used to compensate the effect caused by these movements. For example, iDT solved this by warping the optical flow and estimating the camera motion directly. In the case of Conference dataset, it will be less affected by the camera motion since the camera is set still when capturing the video.

Hence, the effects can be implicitly compensated by subtracting the flow mean on the motion vectors. The final results are reported as shown in the table below.

Freezing Point	Clip-length (L)	Flow Mean Subtraction	Clip Acc.	Video Acc.
None	8	False	73.2%	76.8%
None	16	False	77.8%	81.3%
Conv2_x	16	False	71.3%	72.9%
None	16	True	79.4%	83.7%

**Table 3.3 Results of OF network with different settings**

According to the results, it can be concluded that the OF stream with the best accuracy is configured with  $L = 16$ . In addition, there is no any freezing layer configured in the pretrained model. Thus, the model is able to update its parameters freely. Flow mean subtraction is also applied to obtain better accuracy.

### 3.5.3 Fused Two-Stream Network

Average fusion is applied in order to fuse the both of RGB and OF models with the best performance. Other than clip and video-accuracy, 10-crop testing method is used to further improve the inference results. After inferencing with the Conference Dataset, the result of the fused model is slightly improved. The final accuracies of each stream are as shown in Table 3.4.

Modality	Clip Acc.	Video Acc.	10-crop
RGB	83.8%	89.7%	90.8%
OF	79.4%	83.7%	85.4%
Fused	85.2%	90.1%	91.3%

**Table 3.4 Final performance of RGB, OF and Fused stream**

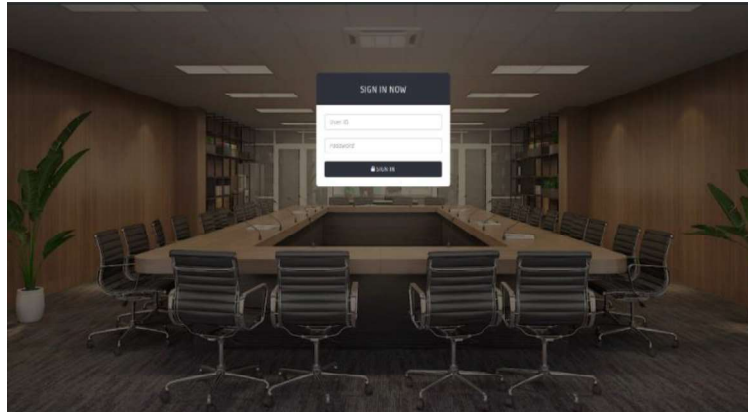
Other than that, an additional experiment is conducted to find out the limiting factor of the model performance. The ‘Non-Meeting’ class is eliminated to test if the model would perform better with only ‘Occupied’ and ‘Non-Occupied’ classes. Eventually, it is discovered that the accuracy is able to reach 99% on the Conference Dataset. This has proven that the ‘Non-Meeting’ class is the major problem that brings down the model performance. Since the class distribution of dataset generated is uneven, data augmentation is applied to further increase the number of samples in ‘Non-Meeting’

class. Still, the model is unable to generalize well due to the insufficient amount of such instances in the dataset.

## CHAPTER 4 WEB APPLICATION DEVELOPMENT

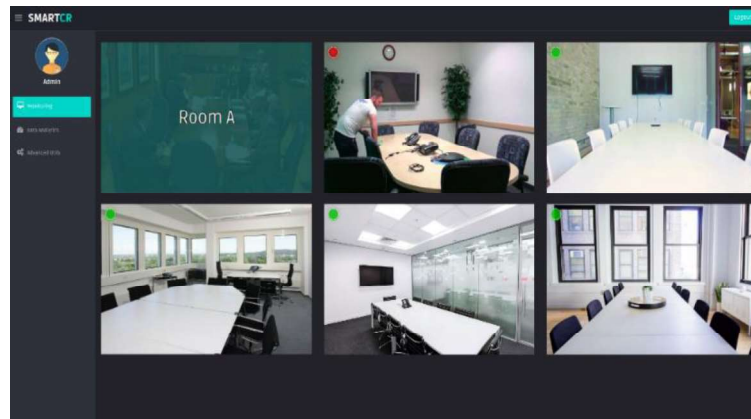
A web application namely SMARTCR has been developed in order to test the capability of the built model based on the experiments conducted above.

**Login Page.** A simple login page is provided for system administrator or other target users to gain access into the application via personal credentials.



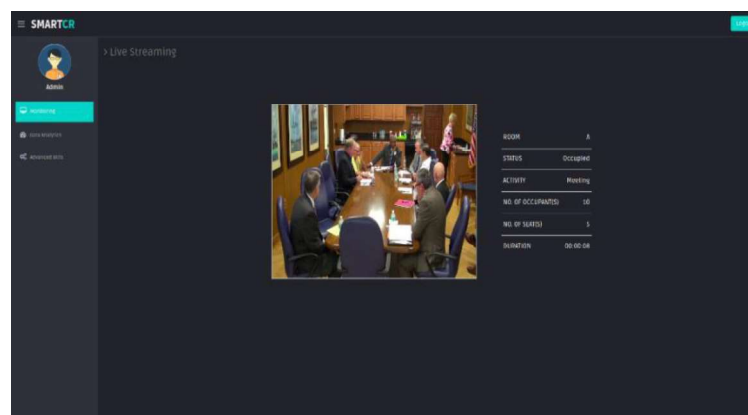
**Figure 4.1 Login page**

**Administration Page.** The user will then be directed to administration page in order to monitor the situations in meeting rooms. With Raspberry Pi cameras installed at 6 different conference rooms, the recorded footages are streamed in real-time and the outputs will be displayed in this page on-the-fly. Instead of showing live recordings, this system will only show only pre-recorded conference videos which were obtained externally for demonstration purposes. First of all, the blinking red/green lights located at the top-left corner of each room panel are actually indicating the availability of rooms. This could directly notify users whether the rooms are occupied or non-occupied. In this case, green lights stand for non-occupied rooms, while red light represents occupied rooms. Unfortunately, performing multiple event detections at different rooms concurrently are impossible for now because the current system is only operated on a machine with GTX 1050Ti GPU. After several attempts, it is verified that only a single event detection and object detection could be performed at the same time before the GPU gets running out of memory. Thus, these pre-recorded videos are halted in the background. The streaming would only be initiated after any mouse-clicked operation on the selected room panel.

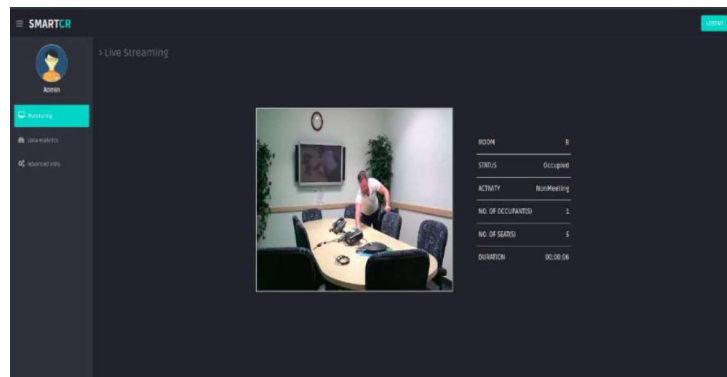


**Figure 4.2 Administration page (Index page)**

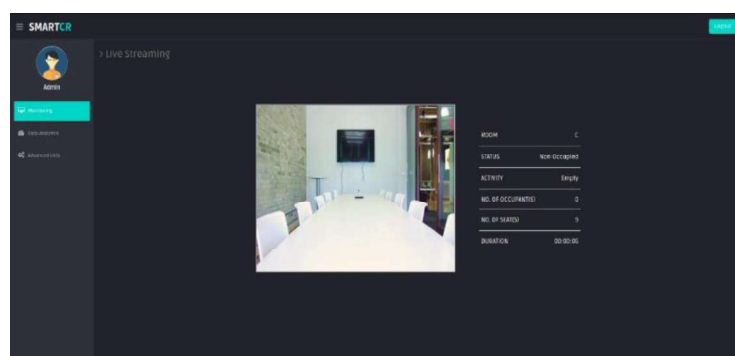
**Monitoring Page.** As mentioned earlier, the event and object detection would only be initiated after any mouse-clicked operation on the selected room panel. This is actually just for demonstration purpose due to the limited computational resource. In actual fact, both event and object detection are able to work on-the-fly for in the real-world application. To illustrate, in order to acquire immediate event detection, 16 frames from the video streams would be drawn into the system in the first place. The exponential weighted average is applied on the predicted scores in order to avoid sudden change of the class predictions. In addition, the object detection will be started on the first frame and will continue to make predictions for every 320 frames (10 seconds). The number of occupants and seats in the meeting room would be displayed on the screen via the prediction retrieval from object detection. The sampling rate would be set to 5 for better sampling after completing the first prediction. The duration would only be calculated once the 'Meeting' event started. 3 of the events including 'Meeting', 'Non-Meeting' and 'Empty' are demonstrated using the diagrams below.



**Figure 4.3 'Meeting' in room A**



**Figure 4.4 'Non-Meeting' in room B**

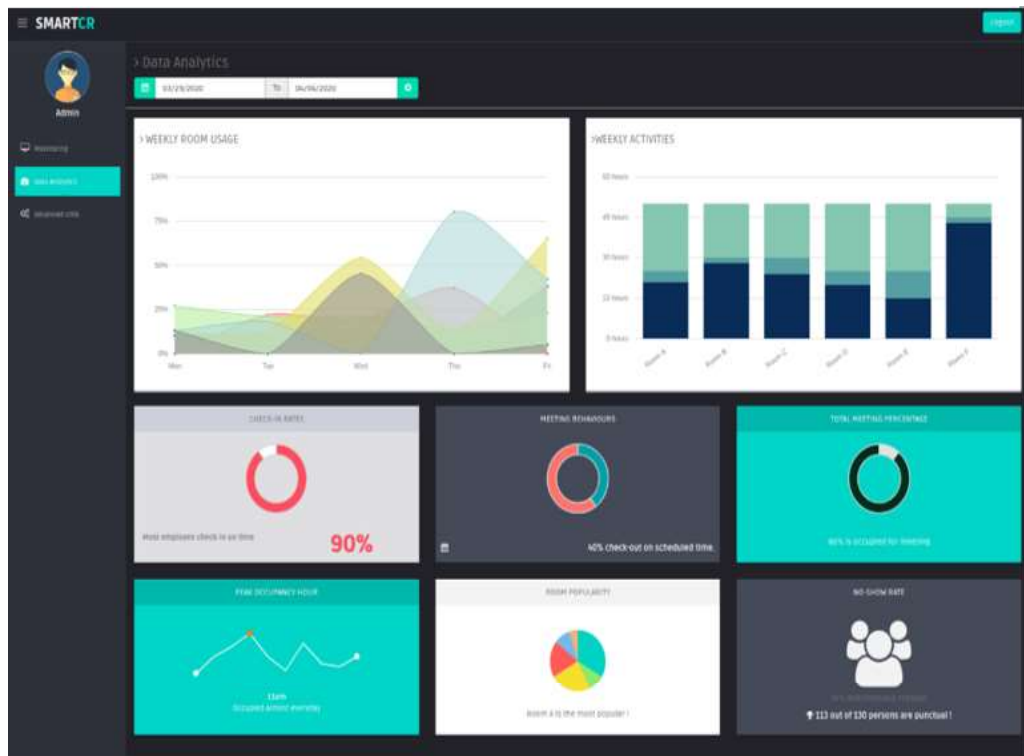


**Figure 4.5 'Empty' in room C**

**Data Analytics Page.** As mentioned earlier, precise analytics or reports could be derived via the information provided in both event and object detections. Thus, this page is created just to showcase the capabilities of this system. On top of the screen, there is an input form to prompt user for their desired date range for the reports. Various reports are displayed in this page using self-generated dummy data. For example,

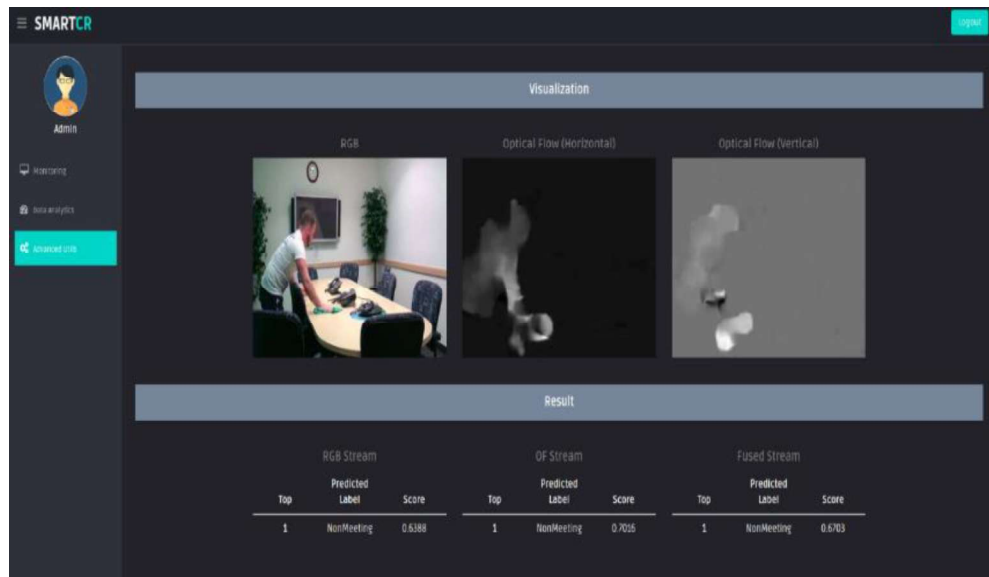
- Weekly room usage: An interactive graph whereby the percentage of the meeting room that is occupied within the selected date range would be displayed.
- Weekly activities: An interactive graph in which the detailed room usage is provided for describing the number of hours spent for each of the events including 'Meeting', 'Non-Meeting' and 'Empty'.
- Check-in rates: A graph to show the percentage of employees checking-in the meeting rooms on scheduled time.
- Meeting behaviors: A graph to show the percentage of employees checking-out the meeting rooms on scheduled time.
- Total meeting percentage: A graph to indicate the percentage of overall 'Meeting' cases within the selected date range.

- Peak occupancy hour: A graph used to indicate the peak timeslot of meeting room usage within the selected date range.
- Room popularity: A pie chart to show the most popular rooms during the selected date range.
- No-show rates: Statistics to indicate the percentage of occupants who are absent for the meeting based on the schedule.



**Figure 4.6 Data Analytics Page**

**Developer Page.** Meanwhile, there is also a page created to showcase and visualize the work flows of event detections in the background. In this page, the RGB, OF (Horizontal) and OF (Vertical) frames are visualized separately. The predicted scores for the RGB, OF and Fused stream are displayed accordingly.



**Figure 4.7 Developer Page**

## CHAPTER 5 CONCLUSION

Event detection in meeting rooms is critically important as one may misuse the conference room by occupying it merely for irrelevant purposes. Although there are plenty of existing systems developed to assist in managing the meeting rooms, most of them are lacking of the ability to recognize the on-going events in conference rooms. Hence, companies are unable to ensure the proper utilization of the conference rooms. In order to solve this issue, a web-based SCR system that is able to detect and classify the events in real-time, is delivered in this project. To perform video classification with higher output accuracy and speed, a robust human action recognition technique, which is two-stream R(2+1)D network architecture is implemented in this system. Before implementing the entire system, a piece of dataset is generated for training the model. By adopting two-stream framework in this project, both the spatial and temporal motion information could be captured and learned at the same time.

In addition, a powerful object detection model, YOLOv3, is applied in this system to act as an object counter for occupants and seats. As such, useful analytics such as weekly occupancy rates would be derived via the information collected. This is vital for companies to gain insights into the usage patterns so that better policies could be imposed for managing the resources efficiently.

However, there are several shortcomings in this project whereby the performance R(2+1)D is limited by the bottlenecks in temporal modelling. Thus, this project could be further extended by adding a temporal fusion using LSTM for modelling longer-term motion information. To illustrate, the outputs from both RGB and OF streams could be fed into LSTM layers right before the FC layer to predict action classes. However, huge and powerful computational resources are required for this work as both RGB and OF streams have to be trained in parallel before feeding to the LSTM layer. Thus, we would need to satisfy the memory requirement before proceeding with this implementation. We hope that this project could be beneficial for corporations to manage their conference rooms much more efficiently.

## BIBLIOGRAPHY

24 Ways Amazon Alexa Skills Can Help Your Small Business Today, 2019. Available at: <<https://smallbiztrends.com/2017/08/amazon-alexa-for-small-businesses.html>>. [15 July 2019].

Amazon Web Services, Inc 2019, *Alexa for Business – empower your organization with Alexa*. Available at: <<https://aws.amazon.com/alexaforbusiness/>>. [15 July 2019].

Afshari, S, Woodstock, T, Imam, M, Mishra, S, Sanderson, A & Radke, R 2015, ‘The Smart Conference Room: An Integrated System Testbed for Efficient, Occupancy-Aware Lighting Control’, *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pp. 245-248. Available from: ACM Portal: ACM Digital Library. [18 July 2019].

Brownlee, J 2019, *A Gentle Introduction to Computer Vision*. Available at: <<https://machinelearningmastery.com/what-is-computer-vision/>>. [1 July 2019].

Carreira, J & Zisserman, A 2017, ‘Quo vadis, action recognition? a new model and the kinetics dataset.’, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299-6308.

Farneback, G 2003, ‘Two-Frame Motion Estimation Based on Polynomial Expansion.’, Paper presented at *Scandinavian conference on Image analysis*, Springer, Berlin, Heidelberg, pp. 363-370.

Finnegan, M 2019, *Alexa for Business: What it does, how to use it*. Available at: <<https://www.computerworld.com/article/3279733/alexa-for-business-what-it-does-how-to-use-it.html>>. [15 July 2019].

Girshick, R 2015, ‘Fast R-CNN.’ *Proceedings of the IEEE international conference on computer vision*, pp. 1440-1448.

Girshick, R, Donahue, J, Darrell, T & Malik, J 2014, ‘Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation.’ *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587.

- Ji, S, Xu, W, Yang, M & Yu, K 2013, '3D Convolutional Neural Networks for Human Action Recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221-35.
- Kay, W, Carreira, J, Simonyan, K, Zhang, B, Hillier, C, Vijayanarasimhan, S, Viola, F, Green, T, Back, T, Natsev, A, Suleyman, M, & Zisserman, A 2017, 'The Kinetics Human Action Video Dataset'.
- Kong, Y & Fu, Y 2018, 'Human Action Recognition and Prediction: A Survey'.
- Koohzadi, M and Charkari, N, M 2017, 'Survey on Deep Learning Methods in Human Action Recognition', *IET Computer Vision*, vol. 11, no. 8, pp.623-632.
- Karpathy, A, Toderici, G, Shetty, S, Leung, T, Sukthankar R, Fei-Fei, L 2014, 'Large-Scale Video Classification with Convolutional Neural Networks', Paper presented at *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Columbus, OH, pp. 1725-1732.
- Prest, A, Schmid, C, & Ferrari, V 2012, 'Weakly Supervised Learning of Interactions Between Humans and Objects.', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 601–614.
- Redmon, J & Farhadi, A 2018, 'YOLOv3: An Incremental Improvement'.
- Ren, S, He, K, Girshick, R & Sun, J 2015. 'Faster R-CNN. Towards Real-Time Object Detection with Region Proposal Networks.' *In Advances in neural information processing systems*, pp. 91-99.
- Saralegui, U, Antón, M, Arbelaitz, O & Muguerza, J 2019, 'Smart Meeting Room Usage Information and Prediction by Modelling Occupancy Profiles', *Sensors*, vol. 19, no.2, pp. 353. Available from: MDPI AG [30 July 2019].
- Sargano, AB, Angelov, P & Habib, Z 2017, 'A Comprehensive Review on Handcrafted and Learning-Based Action Representation Approaches for Human Activity Recognition.', *Applied Sciences*, vol. 7, no. 1, pp. 110.
- Simonyan, K & Zisserman, A 2014, 'Two-stream Convolutional Networks for Action Recognition in Videos', *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pp. 568-576.

- Soomro, K, Zamir, A& Shah, M 2012, 'UCF101: A Dataset of 101 Human Actions Classes from Videos in The Wild'.
- Tran, D, Bourdev, L, Fergus, R, Torresani, L & Paluri, M 2015, 'Learning Spatiotemporal Features with 3D Convolutional Networks.', *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4489-4497.
- Tran, D, Wang, H, Torresani, L, Ray, J, LeCun, Y & Paluri, M 2018, 'A Closer Look at Spatiotemporal Convolutions for Action Recognition.', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6450-6459.
- Wang, H, Kläser, A, Schmid, C & Liu, CL 2011, 'Action Recognition by Dense Trajectories.', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3169-3176.
- Wang, H, Kläser, A, Schmid, C & Liu, CL 2013, 'Dense Trajectories and Motion Boundary Descriptors for Action Recognition.', *International journal of computer vision*, vol. 103, no. 1, pp.60-79.
- Xiao, X, Xu, D & Wan, W 2016, 'Overview: Video Recognition from Handcrafted Method to Deep Learning Method.', Paper presented at the *International Conference on Audio, Language and Image Processing (ICALIP)*, IEEE, China, pp. 646-651.

# Event Detection for Smart Conference Room Using Multi-Stream Convolutional Neural Network

Belinda Khoo Pai Lin, Universiti Tunku Abdul Rahman

## Introduction

Conference rooms are designed to keep highly sensitive information and conversation for facilitating business operations. However, one may misuse the rooms for privacy usage. Thus, event detection in conference rooms is important to assist the organizations in managing the resources in more efficient way.

- **Problem:** Incapability of existing SCR system to recognize on-going events in meeting rooms, which leads to the occurrence of unpleasant scenarios in the workplace.
- **Solution:** A real-time system embedded with human action recognition technique is built in order to detect events in conference room based on the motion information collected. Large amount of meeting footages are collected for generating such dataset. Object detection is also implemented for deriving useful data analytics.

## System Overview

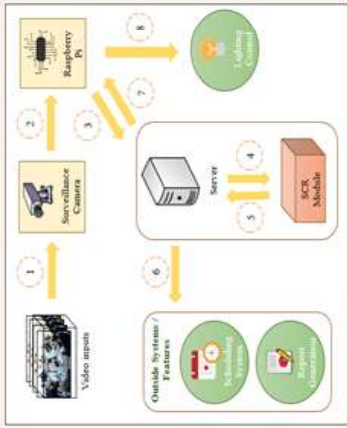


Figure 1 System overview

## Data Preprocessing



Figure 2 RGB, OF (Horizontal), OF (Vertical) frames of 'Meeting'

The video inputs are extracted into a sequence of RGB and OF frames before training. 16 frames are stacked as a short clip.

## Two-Stream R(2+1)D Network Architecture

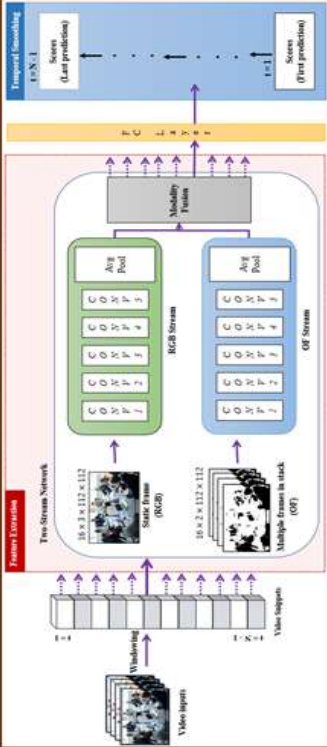


Figure 3 Overview of proposed methodology for SCR system

## R(2+1)D-34 Network Configuration

LAYER NAME	OUTPUT SIZE	R(2+1)D-34
conv1	$1 \times 56 \times 56$	$3 \times 7 \times 7$ , 64, stride $1 \times 2 \times 2$
conv2_x	$1 \times 56 \times 56$	$3 \times 3 \times 3$ , 64, $\times 3$
conv3_x	$2 \times 28 \times 28$	$3 \times 3 \times 3$ , 128, $\times 4$
conv4_x	$7 \times 14 \times 14$	$3 \times 3 \times 3$ , 256, $\times 6$
conv5_x	$1 \times 7 \times 7$	$3 \times 3 \times 3$ , 512, $\times 3$
	$1 \times 1 \times 1$	spatiotemporal pooling, FC layer with softmax

Figure 4 Network Configuration of R(2+1)D-34 [Tran et al., 2018]

## Experiment & Evaluation

**Action Classes:** Meeting, Non-Meeting & Empty.  
**Datasets:** The model is pretrained on Kinetics-400 & finetuned with self-generated Conference Dataset.

### RGB Network:

Feeding Point	Clip Length (L)	Dropout	Clip Acc.	Video Acc.
None	8	-	69.5%	71.3%
None	16	-	74.9%	76.6%
Conv2_x	16	-	78.6%	82.4%
Conv2_x	16	0.5	83.8%	89.7%
Conv2_x	16	0.9	82.3%	87.4%

Figure 5 RGB network with different settings and results

### OF Network:

Feeding Point	Clip Length (L)	Flow Mean Subtraction	Clip Acc.	Video Acc.
None	8	False	73.2%	76.8%
None	16	False	77.8%	81.3%
Conv2_x	16	False	71.3%	72.9%
None	16	True	79.4%	83.7%

Figure 6 OF network with different settings and results

### Fused Two-Stream Network:

Modality	Clip Acc.	Video Acc.	10-crop
RGB	83.8%	89.7%	90.8%
OF	79.4%	83.7%	85.4%
Fused	85.2%	90.1%	91.3%

Figure 7 Final results for each stream

## Conclusion

A real-time SCR system is delivered to detect and recognize on-going events in a meeting room by implementing R(2+1)D network architecture which are robust towards event detection. YOLOv3 is implemented as an object counter. As such, valuable analytics could be acquired for companies to impose better managing policies.

## Reference

Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y. & Paluri, M. 2018, 'A Closer Look at Spatiotemporal Convolutions for Action Recognition', Paper presented at the IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE Salt Lake City, UT.

# PLAGIARISM CHECK RESULT



## Turnitin Originality Report

FYP2\_v3 by Pai Lin Khoo

From FYP submissions (FYP Projects)

Processed on 24-Apr-2020 06:17 +08

ID: 1305656851

Word Count: 13465

Similarity Index	Similarity by Source
3%	Internet Sources: 0% Publications: 1% Student Papers: 2%

### sources:

- < 1% match (publications)

[Joao Carreira, Andrew Zisserman. "Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset", 2017 IEEE Conference on Computer Vision and Pattern Recognition \(CVPR\), 2017](#)
- < 1% match (student papers from 14-Oct-2019)

[Submitted to IIT Delhi on 2019-10-14](#)
- < 1% match (publications)

[Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, Manohar Paluri. "A Closer Look at Spatiotemporal Convolutions for Action Recognition", 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018](#)
- < 1% match (student papers from 03-Mar-2020)

[Submitted to Liverpool John Moores University on 2020-03-03](#)
- < 1% match (publications)

[Heng Wang, Dan Oneata, Jakob Verbeek, Cordelia Schmid. "A Robust and Efficient Video Representation for Action Recognition", International Journal of Computer Vision, 2015](#)
- < 1% match (student papers from 29-Apr-2018)

[Submitted to University of Nottingham on 2018-04-29](#)
- < 1% match (publications)

[Xiaomin Wang, Junsan Zhang, Leiquan Wang, Philip S. Yu, Jie Zhu, Haisheng Li. "Video-level Multi-model Fusion for Action Recognition", Proceedings of the 28th ACM International Conference on Information and Knowledge Management - CIKM '19, 2019](#)
- < 1% match (student papers from 01-Jul-2019)

[Submitted to Hallym University Ilsong Memorial Library on 2019-07-01](#)

<b>Universiti Tunku Abdul Rahman</b>			
<b>Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)</b>			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1 of 1



## FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

<b>Full Name(s) of Candidate(s)</b>	BELINDA KHOO PAI LIN
<b>ID Number(s)</b>	16ACB02442
<b>Programme / Course</b>	BACHELOR OF COMPUTER SCIENCE (HONS)
<b>Title of Final Year Project</b>	EVENT DETECTION FOR SMART CONFERENCE ROOM USING MULTI-STREAM CONVOLUTIONAL NEURAL NETWORK

<b>Similarity</b>	<b>Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)</b>
<b>Overall similarity index:</b> <u>3</u> %  <b>Similarity by source</b> Internet Sources: <u>0</u> % Publications: <u>1</u> % Student Papers: <u>2</u> %	
Number of individual sources listed of more than 3% similarity: <u>0</u>	
<b>Parameters of originality required and limits approved by UTAR are as Follows:</b> (i) Overall similarity index is 20% and below, and (ii) Matching of individual sources listed must be less than 3% each, and (iii) Matching texts in continuous block must not exceed 8 words <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

***Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.***

Signature of Supervisor

Name: Tan Hung Khoon

Date: 24 April 2020

-

Signature of Co-Supervisor

Name: -

Date: -



# UNIVERSITI TUNKU ABDUL RAHMAN


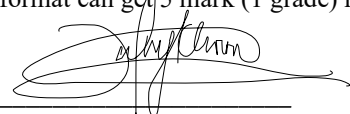
## FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

### CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	16ACB02442
Student Name	Belinda Khoo Pai Lin
Supervisor Name	Dr. Tan Hung Khoon

TICK (✓)	DOCUMENT ITEMS
	Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
✓	Front Cover
✓	Signed Report Status Declaration Form
✓	Title Page
✓	Signed form of the Declaration of Originality
✓	Acknowledgement
✓	Abstract
✓	Table of Contents
✓	List of Figures (if applicable)
✓	List of Tables (if applicable)
✓	List of Symbols (if applicable)
✓	List of Abbreviations (if applicable)
✓	Chapters / Content
✓	Bibliography (or References)
✓	All references in bibliography are cited in the thesis, especially in the chapter of literature review
✓	Appendices (if applicable)
✓	Poster
✓	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)

\*Include this form (checklist) in the thesis (Bind together as the last page)

<p>I, the author, have checked and confirmed all the items listed in the table are included in my report.</p> <p></p> <p>(Signature of Student) Date: 22/4/2020</p>	<p>Supervisor verification. Report with incorrect format can get 5 mark (1 grade) reduction.</p> <p></p> <p>(Signature of Supervisor) Date: 22/4/2020</p>
--	---