SIMULATION-BASED OPTIMIZATION FOR THE PLACEMENT OF SURVEILLANCE CAMERAS IN BUILDINGS USING BIM

Ameen Hamza Albahri

A Thesis in

The Department

of

Building, Civil, and Environmental Engineering

Presented in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy (Building Engineering) at

Concordia University

Montreal, Quebec, Canada

January, 2017

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: Ameen Hamza Albahri

Entitled: **Simulation-Based Optimization for the Placement of Surveillance Cameras in Buildings Using BIM**

and submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY (Building Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final Examining Committee:

Dr. R. Jayakumar	Chair
Dr. Ahmad Jrade	External Examiner
Dr. Govind Gopakumar	External to Program
Dr. Osama Moselhi	Examiner
Dr. Zhenhua Zhu	Examiner
Dr. Amin Hammad	Supervisor

Approved by ______________________________

Chair of Department

March 06, 2017

Dr. Amir Asif

Dean of Faculty

ABSTRACT

Simulation-Based Optimization for the Placement of Surveillance Cameras in Buildings Using BIM

Ameen Hamza Albahri, Ph.D.
Concordia University, 2017

Many companies and organizations highlighted the importance of developing various security monitoring systems for the purpose of protecting their assets against any undesirable events. The process of installing surveillance cameras inside buildings is complex and costly. One of the most important issues that can affect the cost of surveillance cameras in buildings is finding the best placement of cameras. Individuals who are responsible for the camera installation are facing great challenges due to the large number of variables related to this problem. Finding effective scientific methods to address this problem can lead to increasing the efficiency of the camera placement through maximizing the coverage and minimizing the cost. Furthermore, available methods did not take into consideration the impact of different elements (e.g. the HVAC system) that can affect the camera placement.

Building Information Modeling (BIM) is becoming an indispensable process for the Architecture, Engineering and Construction (AEC) industry. In terms of security management, BIM technology can improve the performance of security systems during the design phase because of its ability to identify various elements that surround the 3D surveillance cameras in the form of geometrical and non-geometrical entities.

The objectives of this research are: (1) to develop a method that can help in calculating the coverage of surveillance cameras inside buildings; (2) to investigate the impact of the building elements on the camera configurations, parameters and coverage using BIM; (3) to develop a method that can find the near-optimum types, number and placement of fixed cameras inside a building; and (4) to develop a method that can find the near-optimum placement and movement plan of a Pan-Tilt-Zoom (PTZ) camera inside a building.

A method is developed to calculate the camera coverage inside buildings using BIM and a game engine for the purpose of automating the calculation process and achieving accurate results. This method includes a sensitivity analysis for evaluating the suitable cell size in order to cover the monitored area. Also, the research proposes a novel method using BIM, which provides a new opportunity to better optimize the number and locations of cameras by exploiting the rich information available in the model. The near-optimum results aim to maximize the cameras' coverages and to minimize their costs. BIM is used to define the input of the optimization process and to visualize the results. The method uses Genetic Algorithm (GA) to solve the optimization problem. Finally, the research addresses the placement problem for a PTZ camera. PTZ cameras are used as an addition to fixed cameras in order to ensure the detection of dynamic activities. The research extends the previous method to optimize the placement and movement plan of a PTZ camera inside a building. Several case studies are used to demonstrate the applicability of the proposed methods. The proposed methods can help individuals who are responsible of the camera installation to efficiently determine the near-optimum types, number and placement of cameras needed to monitor spaces in buildings.

ACKNOWLEDGMENTS

I have given all my efforts to prepare this thesis. However, it would not have been possible without the assistance of God and some respectable individuals. I would like to express my sincere gratitude to my adviser, Dr. Amin Hammad for his constant support and intellectual guidance throughout the last four years. I am lucky to be one of his students. I would also, like to thank the members of my committee, Dr. Ahmad Jrade, Dr. Osama Moselhi, Dr. Govind Gopakumar and Dr. Zhenhua Zhu for their valuable inputs and precious time.

I would like to thank my research colleagues for their continuous support to accomplish my PhD degree. I would like to express my sincere gratitude to my friend Dr. Mohammed Mawlana for his intellectual discussion and constructive and valuable comments. Also, I would like to thank my friend Khalid El Ammari for explaining the work of Unity3D which helped in implementing the proposed methods. I would like to express my appreciation to my friend Dr. Faridaddin Vahdatikhaki who assisted me during the planning for this work. In addition, I would like to thank my friends Hassan Siddiqui and Shawkat Ali for explaining some programming concepts and assisted in implementing the proposed methods.

I would like to express my deep gratitude to the Ministry of Education of the government of Saudi Arabia for supporting me throughout my studies. Special thanks and appreciation goes to the Saudi Arabian Cultural Bureau in Canada, which facilitates this educational journey.

Last but most certainly not least, I am grateful and will always be indebted to my parents and brothers for their unconditional love, affection, patience, and encouragements. To my lovely wife Eman and my sweet little sons Hamza and Elyas, who inspired me and provided constant encouragement and patience during the entire process.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF NOMENCLATURE

a	Area
C	Cell
CC_{iv}	Number of covered cells
CP	Cycle path
CT_{p_k}	Coverage-time of the phase p_k
C_{CP}	Coverage of the cycle path
G	Maximum number of generations
g	Generation
H_{max}	Maximum value for the height
H_{min}	Minimum value for the height
IV	Importance value
i	Area number
j	Cell number
L	Maximum value for the length
m	Maximum number of areas
N_x	Cost of camera x
n	Maximum number of cells
n`	Maximum number of covered cells
O	Orientation of the camera
P	Population
p_k	Phase of the PTZ camera
Q	Maximum number of cameras
R	Ray tracing
T	Total time of the cycle path
T_{p_k}	Time of the phase
T_{t_k}	Travel time
U	The total phases' times
W	Maximum value for the width
W_{a_i}	Weight of an important area i
x	Camera number

LIST OF ABBREVIATIONS

AECOM	Architecture, Engineering, Construction, Operation and Maintenance
API	Application Programming Interface
BCLP	Backup Coverage Location Problem
BIM	Building Information Modeling / Model
CAD	Computer-Aided Design
CCTV	Closed-Circuit Television
FOV	Field Of View
GA	Genetic Algorithm
GIS	Geographical Information System
HVAC	Heat, Ventilation and Air Condition
IP	Internet Protocol
IR	Infrared
IT	Information Technology
LAN	Local Area Network
MCLP	Maximal Coverage Location Problem
NP	Non-Deterministic Polynomial Time
PAL	Phase Alternating Line
PICEA	Preference-Inspired Co-Evolutionary Algorithm
PTZ	Pan-Tilt-Zoom
RFID	Radio Frequency Identification
VR	Virtual Reality

CHAPTER 1: INTRODUCTION

1.1 General background and research problem

Supporting security systems of diverse facilities has been given much concern by the public. According to Challinger (2008), an effective security technique should have the ability to specify areas of risks and delineate standards to reduce undesired incidents. For this, most current practices focused on increasing the efficiency of security approaches to ensure the detection of people's movement inside buildings. One way to do this is to increase the number of video surveillance systems in order to obtain as much coverage as possible. However, many organizations are claiming that adopting video surveillance systems is too costly for both installation and maintenance processes (Kelly, 2013). According to a case study implemented by Reid (2005), installing 45 cameras, multiplexers, monitors, recorders and cables to ensure monitoring of several areas inside a facility costed $420,000. It is necessary to know that once the surveillance cameras are placed inside the building, it is too costly to change their positions (Chen et al., 2013). In most situations, surveillance cameras are installed based on experience, which usually is not efficient since there is no scientific method that can ensure the coverage of important areas at a minimum cost (Amriki and Atrey, 2012).

One significant factor that can affect this issue is the process of evaluating the near-optimum number, types and placement of the cameras. Generally, most of the current techniques are in 2 Dimensional (2D) or 3 Dimensional (3D) forms, which showed the following problems. First, the usage of the 2D technique may increase the possibility of not covering some important areas which are located under the Field of View (FOV) of the camera or in places where there are different ceiling or floor elevations. Second, traditional 3D techniques do not consider some building elements that can affect the camera coverage. To illustrate, teams who are responsible for the camera installation usually follow a camera placement layout in order to place the required cameras at their specified locations. Sometimes, these locations are not suitable because of the incomplete coverage caused by geometrical constraints (e.g. ventilation systems and ceiling signboards) which are hard to be detected using traditional 2D or 3D techniques. As a way to avoid these elements, installers usually prefer to change the placement, orientation or even the height of these cameras, which will affect their coverage.

Evaluating the efficiency of the camera coverage requires considering three major elements: (1) the relationship between the used cameras and their surroundings, specifically the important areas; (2) the type of the used cameras including their features, capabilities and limitations; (3) the physical obstacles, such as the existence of different ceiling or floor elevations and the interferences with other building systems; and (4) the non-physical factors that can limit the operating process of these cameras, such as the existence of privacy areas. Although some practices, such as Chen et al., (2013), proved the ability of BIM to detect previous elements, more efforts are needed to investigate the role of this technology in optimizing the placement of surveillance cameras inside buildings. Also, the efficiency of the camera layout can be determined by evaluating the coverage of each camera type. This evaluation is affected by the following factors: (a) the purpose of usage which includes security, safety or work monitoring; (b) the economic features including camera purchasing, installation and maintenance costs; (c) the special conditions including important and unimportant areas, privacy areas and physical constraints; and (d) camera parameters that include Field of View (FOV), focal length, camera height, location and orientation while considering the target height and distance.

Furthermore, individuals who are responsible for the camera installation inside buildings are facing great challenges due to the large number of variables related to this issue. For this reason, researchers developed different techniques to optimize the placement of surveillance cameras inside buildings. However, these techniques did not consider the existence of the above mentioned elements (e.g. ceiling signboards), which may affect the visibility of the placed cameras. More attention is needed regarding the features (e.g. FOV) of the placed cameras and their impact on the camera optimization process. Another important issue is that almost all previous studies implemented their methods on surfaces without considering geometrical constraints, which surrounded the placed cameras and can affect their coverage. This work will benefit from BIM in determining the locations of these constraints and considering them in the camera placement process. Also, the placement of some types of cameras, such as the Pan Tilt Zoom (PTZ) cameras, is much harder to be optimized. To illustrate, most current practices have focused on improving the monitoring, tracking and synchronization processes of these cameras (Chae and Kano, 2007; Yang et al., 2010; and Starzyk and Qureshi, 2011); and ignored their placement inside buildings. This is due to the complexity of the dynamic coverage of PTZ cameras which aims to monitor the movement of people.

This research aims to address the camera placement issue by answering the following questions:

(1) How to calculate the coverage of a single surveillance camera considering different building elements that can affect the camera coverage? How to optimize the placement of this camera considering related factors?

(2) What are the factors that can affect the coverage of multiple cameras? How to optimize the placement of multiple cameras?

(3) How to improve the proposed methods in order to consider different camera types and placement?

1.2 Research objectives

The main target of this research is to develop a novel and scientific method that can help in placing surveillance cameras inside buildings. The specific objectives are:

(1) To develop a method for calculating the coverage of surveillance cameras inside buildings.

(2) To investigate the impact of the building elements on the camera configurations, parameters and coverage using BIM.

(3) To develop a method that can find the near-optimum types, number and placement of fixed cameras inside a building.

(4) To develop a method that can find the near-optimum placement and movement plan of a PTZ camera inside a building.

1.3 Research significance

This research is very beneficial for individuals who are responsible for installing cameras inside buildings. The research aims to achieve accurate coverage results and near-optimum camera placement using scientific approaches instead of depending on human experience. Using the research methods, users will be able to determine the near-optimum types, number and placement of cameras needed to monitor an area inside a building. The proposed methods will ensure the following: (1) to obtain the maximum coverage of surveillance cameras; (2) to reduce the cost of coverage by reducing the number of placed cameras of suitable types; and (3) to ensure adequate coverage in case of using PTZ cameras.

1.4 Research organization

This chapter is presenting an introduction about the issue of the camera placement inside buildings and the considerations that affect its performance. Furthermore, the current status of relative studies along with their limitations was briefly discussed. Also, the research problem, objectives and importance were addressed. Chapter 2 includes a comprehensive literature review. Chapter 3 provides an overview of the research methodology. Chapter 4 explains the first method of calculating the camera coverage in buildings using BIM. Chapter 5 explains a method of simulation-based optimization for the camera types, number and placement in buildings using BIM. Chapter 6 explains a method of simulation-based optimization for the placement of a PTZ camera in building using BIM. Chapter 7 discusses the conclusions and future work.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

The purpose of this chapter is to review the research and the latest practices related to camera placement optimization. Also, it aims to highlight the issues and standards that can affect the study's objectives. The chapter reviews the following topics:

(1) The configurations, parameters and features that affect the selection and placement of cameras.

(2) The rules that should be considered during the camera placement process.

(3) The role of BIM in addressing several issues in the AECOM industry and close to the camera placement issue.

(4) The existing optimization methods that can assist the research methodology.

(5) The most related practices that focused on the camera placement.

2.2 Security monitoring

According to Wilkis (2015), there is no system that can ensure complete coverage, and every system has its own strong and weak points. Focusing on security systems of multiple surveillance cameras, the cameras may leave small uncovered areas which might be a target for suspicious activities (Gigliotti and Jason, 1984). Therefore, many approaches have been developed for strengthening surveillance systems. A report by Challinger (2008) listed some of these methods used inside buildings as follows: access control, security officers, emergency plans, physical security, closed-circuit television (CCTV), lighting, documented procedures, and security awareness activities. The report also pointed out that crimes, disorder, emergencies, and fear are the most significant issues that concern security officers. These issues can be classified as follows (Challinger, 2008):

(1) Crimes

Crimes can be classified as: (1) stealing incidents that may occur in either private locations or public ones such as car parking areas, (2) damaging and destroying public assets, (3) violent and offensive behaviour against the community, and (4) unauthorized access to prohibited areas by intruders who may conduct undesirable activities such as stealing or spying.

(2) Emergencies

These are incidents that affect the stability of the environment or the community and that need quick responses by professionals. According to Challinger's report, these incidents mostly occur because of the following two reasons: (1) incidents caused by humans, such as fire, infrastructure breakdown and terrorist attacks, and (2) incidents caused by nature, such as critical weather conditions.

The report also outlined that in order to achieve the optimum security means, it is necessary to determine the weak points, and the rules that can mitigate the impact of these points. In case of the unavailability of such rules, security experts should establish new rules and procedures that are suited to their security problem. The report also added that high-rise buildings have some features that make them more difficult to secure than other building types. These features are: (1) restricted entries that make the evacuation process more difficult; (2) the need to use elevators to reach high levels in most cases of emergency; (3) the existence of some public services inside these buildings, such as metro stations, making the security monitoring much harder to control; and (4) vast amounts of people, which makes it hard to distinguish residents from offenders.

2.2.1 The role of architectural design in supporting security monitoring

Design is one of the most significant factors that can reduce weaknesses of security systems inside buildings. According to O'Block (1981), the role of environmental design in mitigating crime incidents was not comprehended until the beginning of the 1960s. During this period, a housing project was deliberated between two architects, members of the St. Louis Police Department and two sociologists. They came up with a new concept named "the defensible space", which aims to support the security of property by using tools such as design, increasing surveillance and improving safety.

O'Block (1981) highlighted the importance of considering additional courses, concerned with the issue of crime mitigating methods, in the curriculum of architecture and urban design schools, for the purpose of meeting the demands of a safe environment. The study also listed six recommendations for supporting security monitoring, established by the National Institute of Law Enforcement and Criminal Justice (1974) as follows: (1) increasing surveillance areas; (2) separating private areas from public areas; (3) supporting the concept of territoriality; (4) monitoring access to spaces; (5) separating inconsistent activities; and (6) creating alternatives that can decrease the possibility of crime.

Also, the study recommended the integration of efforts among different groups, such as architects and security employees, in order to strengthen the final design from the aspect of security. The study mentioned that this integration can decrease the need for expensive adjustments that usually occur due to inefficiency in the planning stage and after the construction process is completed. This would increase the need to ensure interoperability among different parties that share the same project. As a matter of fact, clients should not expect that architectects will inevitably consider security technologies in the design of their projects, because architects assume that their clients are aware of these technologies (O'Block, 1981). This relationship forms the basis of the research issue that highlights the importance of having a robust communication method during the process of deciding the appropriate camera layout.

It is also necessary to use tools that can ensure adequate data gathering during the process of designing the appropriate layout of surveillance cameras. According to Kruegle (1992), architects should consider a surveillance cameras checklist that can provide beneficial information to the client as per the following: (1) the quality of surveillance camera; (2) the type of lens and monitors that will fulfill the needs; (3) information about the group that will be responsible for controlling the monitors; (4) technical information about the electrical outlets of the camera; (5) specifying the maintenance group; and (6) determining the geometrical features of the control room.

2.2.2 Issues in high-rise buildings

Different security issues can affect different types of buildings. For the sake of this research, the focus will be on high-rise buildings, since they include more difficult conditions affecting security monitoring systems than other building types. The report of Challinger (2008) clarified that high-rise buildings have features that can generate particular security issues. The following are some of these features and their associated challenges:

(1) Size within the building

High-rise buildings can be exposed to threats due to their huge size. The bigger the size of the building, the more spread out are the spaces and the more difficult to control is the monitoring process. As a result, the process of detecting strangers among all the people is more complicated inside these buildings. In this situation, most of those people have authorized access or might be visitors, but among them could be a stranger that can be considered a threat in the worst case scenario.

(2) Physical features of the building

One of the significant factors that make high-rise buildings more exposed to crime is the physical formation of some of the internal components, such as the elevators and stairs (Spanier and Fishel, 1973). These areas can create blind spots that can be exploited by intruders for the purpose of carrying out illegal activities. According to Challinger (2008), stairwells in high-rise buildings can offer extra locations for robberies and escape ways for criminals. Other public areas such as lobbies and corridors are also exposed to illegal activities.

(3) People access in high rise buildings

The report of Challinger (2008) also clarifies that most high-rise residential buildings nowadays include security control systems such as access control cards and personal identification numbers in order to limit accessibility to authorized individuals. In addition, guests accessing high-rise residential buildings are monitored by surveillance cameras to allow residents to distinguish their faces (Challinger, 2008). In Korea, robbers frequently present themselves as maintenance groups and convince inhabitants to allow them inside their building. A solution was established by using a fingerprint reader system, which installed inside the door. After entering the suitable Personnel Identification Number (PIN) to a keyboard, the fingerprint reader will slide out for the visitor. This reader can detect the visitor's identity through his fingerprint. The identity data is sent to the company, which invents this tool (Hi-Tech High-Rise, 2002).

(3) Physical security tools

The report also listed a number of physical security tools used in high-rise buildings such as locks, alarms, lighting, perimeter gates and fences. These tools were established in order to make the accessing process much harder for strangers and criminals. In other words, physical security methods can reduce the space that the criminals might use to enter the building. Even though these tools might seem beneficial in hindering criminals, other parties perceive the development of these security tools as a challenge to their abilities. In some cases, increasing the usage of physical security tools can make inhabitants feel uncomfortable (Challinger, 2008).

(4) Surveillance cameras

According to Kruegle (1992) the concept of surveillance cameras is related to the placement of cameras at specific locations (mostly ceilings) where they face some targets, while the control room can be located far away (hundreds or thousands of feet) from the cameras. This system usually includes: camera lenses, coaxial cable or wireless transmission means, a monitor, and a video recording system. Challinger (2008) also outlined that surveillance cameras have been extensively applied in high-rise buildings. Furthermore, Internet-based surveillance cameras offer the chance for people to see what is happening to their assets when they are outside the building. These cameras provide other features such as motion detectors, which can activate alarms in case of unauthorized intruders. Also, these cameras allow storing numerous digitized videos in their system. However, surveillance cameras can sometimes pervert the security endeavors. For example, one significant objective of using these cameras is to store historical data that can be of particular help for an event investigation, but most of the time these data are just records of some previous infractions not related to the event (Challinger, 2008).

(5) Security Staff

Most high-rise commercial buildings use formal security officers to ensure crime limitation. That means the existence of such staff can prevent criminals from continuing their illegal actions and can increase the occupants' confidence. These officers can perform various tasks such as managing access inside buildings, overseeing car access, or managing the control rooms of the surveillance system. Other employees inside the monitored building can also share the role of ensuring security aspects for this building (Challinger, 2008). A study by Stellitano (2005) indicated a training program that trains doormen to support the security management process by knowing the way to deal with some incidents, such as detecting strange and unusual individuals or actions.

2.2.3 Types of surveillance cameras

(1) Fixed and PTZ cameras

Fixed cameras are placed in a static position so that they can ensure a constant coverage to a specific area. The view of a fixed camera provides the same scene until it repositioned. The resulted scenes are recorded and instantaneously seen on a monitor by security individuals (Green, 1999). An article by Kumar (2012) clarifies that the reason for naming the Pan Tilt Zoom (PTZ) camera is its capability to perform pan rotation from left to right, and tilt rotation from up to down, in addition to the feature of zooming in and out from a specific view. More specifically, the movements that the PTZ camera can do are (Cassidy, 2009):

(a) Pan movement: the movement of the camera head from right to left in a spherical shape;
(b) Tilt movement: the movement of the camera head from up to down, while maintaining the horizontal axis (of the pan movement); and
(c) Zoom: the process of altering the focal length in order to make the target look closer to, or further from, the lens by way of increasing or decreasing the pixels of the picture.

These movements are beneficial in places where people's movement is expected from different directions, such as corridors. These types of cameras can help in tracking people movement, since their movements are not fixed like other types of surveillance cameras. These types of cameras can be controlled and operated by using a simple keyboard and receiver.

A contemporary improvement of conventional PTZ cameras, as shown in Figure 2-1, is the prevalence of PTZ dome cameras. The operating process of this type is the same as the conventional one except that all the basic components of the pan and tilt head and the control electronics are located inside the dome itself. These domes can serve different purposes due to their various surface types, from transparent, semitransparent to opaque. PTZ dome cameras, as shown in Figure 2-2, are more beneficial than conventional ones in several aspects. These domes can easily rotate in all directions, with 360-degree rotation for the pan movement and the ability to rotate in the tilt direction. They have a feature named "auto-flip", which allows the camera to automatically follow any object passed beneath it. Unlike conventional PTZ cameras that have stopping movement point, the auto-flip feature gives the ability to follow an object beyond the stopping point in a circular manner and in any direction (Kumar, 2012).

Figure 2-1 Conventional PTZ camera (Kumar, 2012)

In addition, Internet Protocol (IP) PTZ cameras have an IP address built inside them, allowing the monitoring process to be started from any place that uses a standard web browser. To operate the IP PTZ cameras, users can easily connect the camera to a network and can gain the benefit of much popular software, based on features included in the selected IP PTZ cameras (Kumar, 2012).

Figure 2-2 PTZ dome camera (Kumar, 2012)

The main advantages and disadvantages of the PTZ cameras are summarized in Table 2-1 (Anukul, 2015).

Table 2-1 Advantages and disadvantages of the PTZ cameras

Advantages	*Disadvantages*
• Can be manually used by operators for tracking and zooming purposes. • The ability to automate the rotation (i.e. pan and tilt movements) of the camera from one target to another. • The ability to track targets while monitoring large areas. • The ability to integrate these cameras with alarm sensors, which can automatically warn the camera to monitor a specific target (e.g. intruder).	• Can only record areas where the camera is directed. This can result in missing important incidents occurring in other areas. • Have complex mechanical devices and shorter life span compared to fixed cameras. • Have smaller sensors than fixed cameras in order to fit with other mechanical devices. Therefore, PTZ cameras in general have poorer image quality than fixed cameras. • Are very sensitive to vibration, which can cause unsatisfactory videos.

One distinctive feature in PTZ cameras is the privacy masking tool (Kumar, 2012). This offers the capability to cover specific regions in a camera's view, as shown in Figure 2-3. The location of the mask can update itself to run simultaneously with the movements of the camera in order to make sure that the privacy area is not being disclosed because of the movement. This feature will enable users to avoid being sued because of invasion of privacy.

Figure 2-3 Masking feature in PTZ cameras (Kumar, 2012)

(2) Infra-Red Cameras

This type of cameras is necessary in case of lacking of adequate light source. An infrared (IR) camera usually operates as a regular camera in the daytime. At night, these cameras generate black and white scenes where a regular colored camera is not efficient (Cohen et al., 2009). An IR camera includes an inside device named image intensifier, which is sensitive to invisible near infrared radiation. Image intensifier is able to magnify the available light to obtain visions. This device is integrated with the infrared illuminator to capture clear night visions (Electrophysics Introduces, 2006). Also, these types of cameras provide weak color readings during the day, as shown in Figure 2-4. For this, it is advisable to attach an infrared filter during the daytime as a solution to enhance these readings (Cohen et al., 2009).

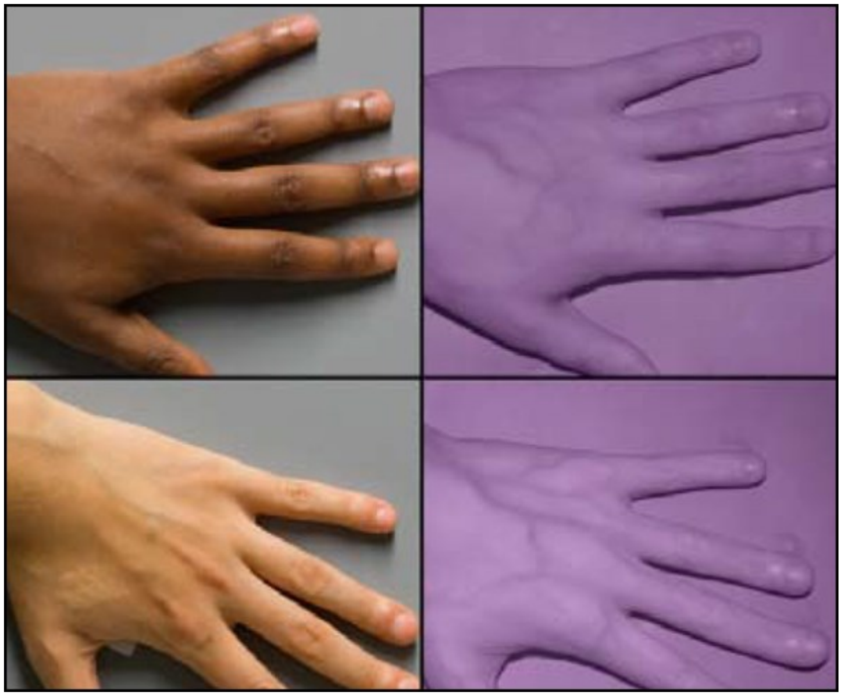


Figure 2-4 Color reading of infrared camera (Cohen et al., 2009)

(3) IP and analogue cameras

Analogue cameras transmit their videos in a conventional format through a cable to a digital video recorder (DVR) to be stored in a hard drive. Before this transmission, videos are converted to analogue in order to be received by the analogue DVR. IP cameras can gather image data the same way as analogue cameras, except that they transfer the captured images digitally through a Local Area Network (LAN) or by the Internet. Using these cameras will offer the following benefits: (1) the ability to see the transmitted images from far places; and (2) the opportunity to provide greater quality images, since it is not restricted by the conditions of the Phase Alternating Line (PAL) standard (Cohen et al., 2009). There are many types of IP cameras with different resolutions and usually referred to as megapixel cameras (Cohen et al., 2009). Some types of surveillance cameras are presented in Appendix A.

2.2.4 Considerations related to surveillance cameras

(1) Scene contrast

The concept of scene contrast can be explained as follows: (1) when maintaining specific levels of light passing through a scene, adequate contrast can be obtained; and (2) mixing high levels of light can create too much contrast and cause unclear photos. If high and low light levels are encountered in the same view, the resulting contrast is usually too high to be managed by the camera. This can cause an impact that shows a subject as a dark shadow or the surroundings are invisible, as in Figure 2-5. In addition, in some cases where an object passes under a light source, the upper part of the object becomes considerably brighter, and when it passes on, it converts to dark again. This issue can be solved by applying a diffuser to the camera or by establishing more lighting in areas that include a high density of people (Cohen et al., 2009).

(a) High contrast effect (b) Low light level

Figure 2-5 Scene contrast (Cohen et al., 2009)

(2) Lens / aperture

The lens is the core element that controls the visibility of the surveillance cameras. Kruegle (1992) explained the process of camera visibility as follows: (1) a scene is illuminated by a light source; (2) a portion of this light is reflected to the camera where it will be gathered by the lens; (3) this lens will focus the scene to a part called the image tube. With the assistance of the camera electronics, the tube will convert the image into signals to be transferred to the monitor part. The lens also plays a vital role with the camera sensor in controlling and directing the field of view generated by the camera, which varies from wide angle to narrow angle. Another important element is the aperture, which has the ability to collect an amount of light to be transmitted to the sensor part of the camera. It is important to know that the wider the camera aperture, the less field of view the camera will obtain and vice versa (Cohen et al., 2009).

(3) Field of View

The field of view of a camera is the three dimensional region which is occupied by the camera's view. It can be also represented as the viewing property. This region includes all objects, which will be captured and recorded by the camera. Basically, the field of view has a rectangular pyramid shape, which formed by two horizontal and two vertical planes (Douglas and Kerr, 2006), as shown in Figure 2-6. There are three factors that determine and control the FOV of a camera: the lens size, the internal sensor and the distance between the camera and the targeted view (Cohen et al., 2009). Kruegle (1992) defined FOV as "the scene that you can see". This scene is composed of width, height and space between the surveillance camera and the scene. The previous two definitions of FOV can be integrated to form the following definition: the hypothetical coverage shape that emits from the camera lens and has two horizontal and vertical faces, and which has the ability to locate a target inside the pyramid shape. Also, it is necessary to take into consideration that any FOV needed for a camera should be kept away from areas that include shadows and blind spots (Cohen et al., 2009).

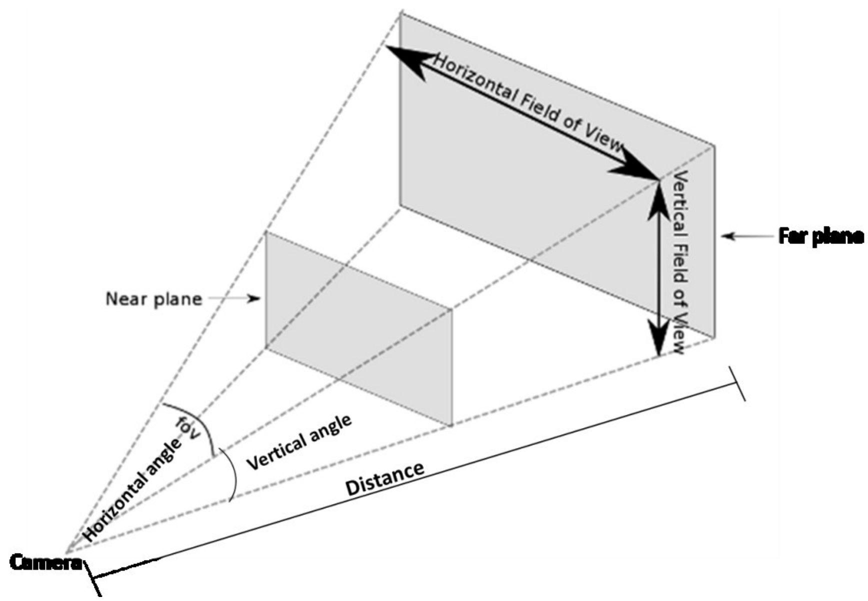


Figure 2-6 FOV concept (Dirksen, 2013)

2.2.5 The effect of privacy on camera placement

A report by the Office of the Privacy Commissioner (2009) highlighted a number of privacy guidelines before installing surveillance systems as the following:

(1) Determining the necessity of surveillance cameras for the case by specifying:

- The purpose of using surveillance cameras.
- The problem to be addressed by surveillance cameras.
- The way surveillance cameras can help in addressing the problem.
- Other options instead of surveillance cameras.

- People that can be consulted to solve this problem.

(2) Considering a clear plan by:

- Developing a business plan for the surveillance system. The plan should include the following: (1) the purpose of using the system; (2) the expected results from the system; (3) the technologies that will be used; (4) people who will operate the system; and (5) ways to decrease privacy effects.
- Consulting experts for the plan.
- Assigning a person for operating the surveillance system.
- Setting policy for the way camera images will be handled.
- Staff training procedures should be included in the plan.

(3) Choosing and placing cameras taking into account the following:

- Selecting the suitable equipment that can achieve the purpose without intruding the privacy.
- Finding technologies that can ensure privacy (e.g. masking feature).
- Placing the cameras in a way that will not affect people privacy.

The Freedom of Information and Protection of Privacy Act (2004) published a guide that outlined a number of considerations before placing surveillance cameras. Some of these considerations can be listed as the following: (1) Surveillance cameras should be placed in specified public areas where there is a necessity for them; (2) Surveillance cameras should not be placed to monitor other external buildings unless there is a need to ensure safety for external possessions; (3) Avoid the placement which monitors privacy places (e.g. washrooms); (4) Cameras may be placed to monitor some exceptional places such as children's' washrooms in order to ensure children safety against any possible criminal activities; and (5) The usage of surveillance cameras should be limited to periods that have high crime rates.

(4) Ensuring people awareness of the existence of surveillance cameras by:

- Placing signs that inform people about the existence of surveillance cameras.
- Signs should provide information about the owner and operator of the surveillance camera, and their contact details.
- Ensuring a full privacy notice in the website or as a hard copy on the reception desk.
- Ensuring that the staff can answer the people about any questions regarding the installed system.

(5) Limiting the collection to important images by limiting the operating hours of surveillance cameras to the necessary periods (e.g. opening hours).

(6) The usage of images should consider:

- Taking rational steps to check if the images are precise, comprehensive and applicable before using them.
- The usage of images should be just for achieving the original purpose.
- Disclosing images should be after having consent from the individual(s) or permission from the police.

(7) Image storing and retention. This includes:

- Ensuring the protection of images.
- Images should be stored for a specific time that serves the main purpose.

(8) Controlling the accessibility for the images by:

- Limiting the accessibility to just the authorized members.

- Developing procedures that can help individuals in accessing their images which were taken by the surveillance cameras.
- Developing procedures of releasing images to the police.
- Keeping a record of images and cases which were accessed by external party.

(9) Auditing and evaluating surveillance cameras by:

- Gathering statistics about the performance of surveillance cameras.
- Conducting annual evaluation of the operation of the surveillance cameras to assess their viability.
- Auditing all related equipment to ensure smooth operation of the surveillance cameras.
- Checking that the staff is following the policies and procedures.

2.2.6 Considerations related to camera placement

Since the main focus of this research is on addressing the issue of camera placement, it is necessary to review a number of considerations that can affect this matter. A report by CheckVideo (2014) highlighted some of these considerations as follows:

(1) Camera location

Camera location is where the installer decided to place the camera inside or outside the building. Usually this decision is affected by three main factors as follows (CheckVideo, 2014):

(a) Camera angle: Basically this is related to the degree of tilt angle (in fixed cameras). For instance, in the case of a camera placed on the ceiling with a tilt angle of 90°, this means the camera is facing the ground, while in the case of a tilt angle of 0°, the camera direction is parallel to the ground, as shown in Figure 2-7. Usually, in order to obtain an efficient view, the tilt angle should be between 15^o- 45^o.

(b) Camera height: To achieve optimal camera location, the camera should be at a height of between 8 and 20 feet.

(c) Camera distance: This distance is affected by the focal length and the target which the camera is trying to detect. Table 2-2 shows different focal lengths when the camera is mounted at a height of 10-20 feet.

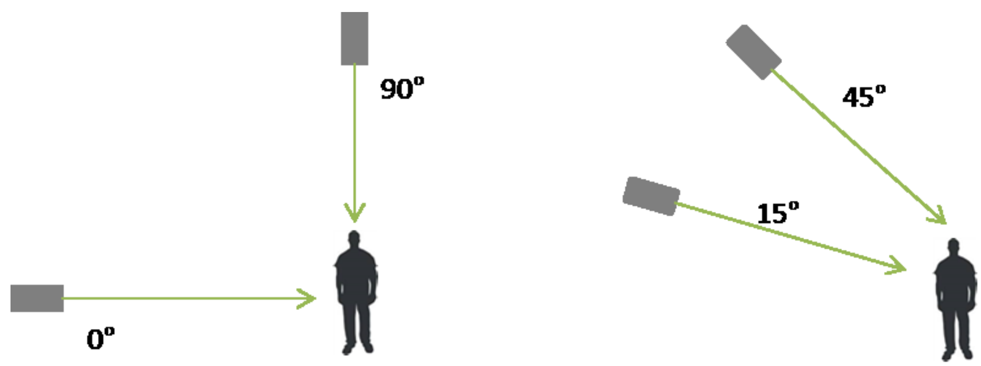


Figure 2-7 Different camera angles (CheckVideo, 2014)

Table 2-2 Areas of person detection (CheckVideo, 2014)

Focal length (mm)	Detection area length (feet)	Min. width of detection area (feet)	Max. width of detection area (feet)
2.8	5-35	15	50
3.6	10-40		
6	15-65		
8	20-85		
12	35-130		

(2) Motion detection

Suitable outcomes will be achieved if the camera is mounted in a way that the target moves across the FOV of the camera, not directly towards it or away from it. Different areas of detection are also shown in Table 2-2.

(3) Camera orientation

The report recommends that the camera should be placed in an upright position, as shown in Figure 2-8, taking into account that any rotation more than 5° will lead to undesirable scenes, as shown in Figure 2-9.

Figure 2-8 Camera with upright direction (CheckVideo, 2014)

Figure 2-9 Undesirable scene (CheckVideo, 2014)

(4) Ensuring secure mounting

It is necessary to install the camera in an appropriate and steady position in order to avoid affecting the camera body or its visibility. Mounting loosely can cause changes in the camera orientation over time because of structural vibrations or other environmental causes.

(5) Illumination

Installers should consider adequate lighting before deciding the placement of the surveillance camera, in order to avoid inefficient visibility, as shown in Figure 2-10. The report gives an example of an adequate degree of lighting such that an individual can read a newspaper inside a room. Also, the report advised not to place the camera close to windows or any lighting source, in order to avoid contrast interference, as shown in Figure 2-11.

Figure 2-10 Inefficient visibility (CheckVideo, 2014)

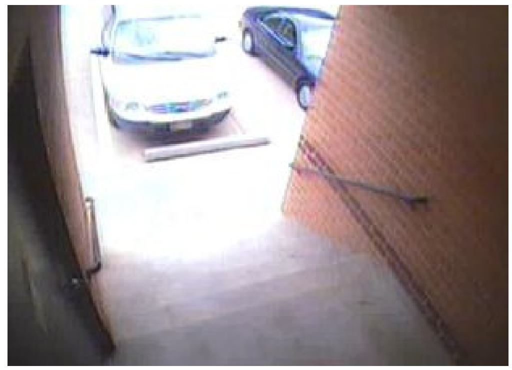


Figure 2-11 Contrast interfering of the camera (CheckVideo, 2014)

In a case where obtaining adequate lighting is not possible, an additional lighting source should be taken into account. Cameras with an infrared feature are beneficial tools that can address such cases. The report recommends the usage of IR cameras with a wavelength of 850 nanometers. The following are some considerations for IR cameras:

(a) The IR camera should be mounted at least 12 feet off the ground in order to avoid reflections.

(b) IR source should be directed at the same point as the camera is directed.

(c) The IR source should be placed one foot above the camera and not below it.

(d) It is important to avoid any surfaces that can cause reflections, such as lamp posts and other structures.

(6) Some effects to avoid

The report recommends avoiding objects that can obstruct the camera view, such as furniture, walls and columns, as shown in Figure 2-12.

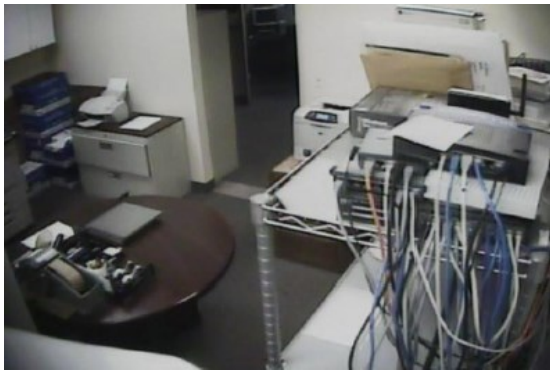


Figure 2-12 Object obstructing the camera view (CheckVideo, 2014)

The report also highlights the importance of taking into account (before placing the surveillance camera) the avoidance of places that might include moving objects such as swinging doors, ceiling fans, flags, water fountains or even insects (for outdoor visibility) which can affect camera visibility, as shown in Figure 2-13.

Figure 2-13 Obstructing the visibility by moving objects (CheckVideo, 2014)

Also, it is important not to place the surveillance camera behind any lighting sources, as shown in Figure 2-14, in order to avoid reflection, glare or extreme lighting, which might affect camera visibility.

Figure 2-14 Effect of placing the camera behind the lighting source (CheckVideo, 2014)

The report also highlighted the importance of considering the effect of reflected surfaces such as mirrors, windows or polished floors, as shown in Figure 2-15.

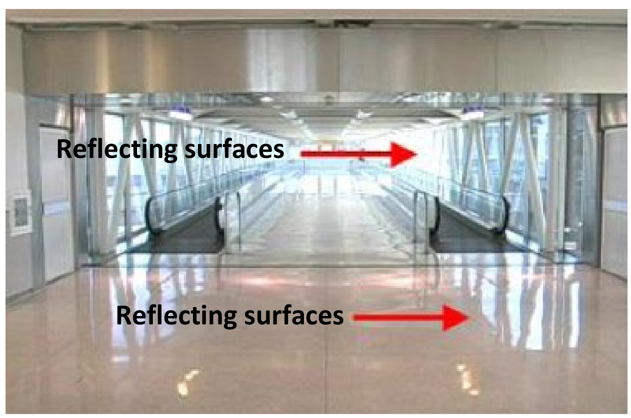


Figure 2-15 Effect of reflecting surfaces on the visibility (CheckVideo, 2014)

Furthermore, the report provided some recommendations that help to monitor entrances. One basic rule is not to point the camera directly at the door because targets will have the opportunity to avoid the camera, as shown in the Figure 2-16. Instead, the focus should be to monitor the route to the entrance and not the actual door, as shown in Figure 2-17. Lastly, the report highlights the importance of reducing the view of swinging doors.

Figure 2-16 Pointing the camera directly to the doors (CheckVideo, 2014)

Figure 2-17 Directing the camera to the route of the entrance (CheckVideo, 2014)

In case of monitoring storage spaces, the focus should be on maximizing the view and avoiding doorways, moving objects or flashing lights, as shown in Figure 2-18. Also, it is necessary that individuals appear vertically in full view for at least one location, as shown in Figure 2-19.

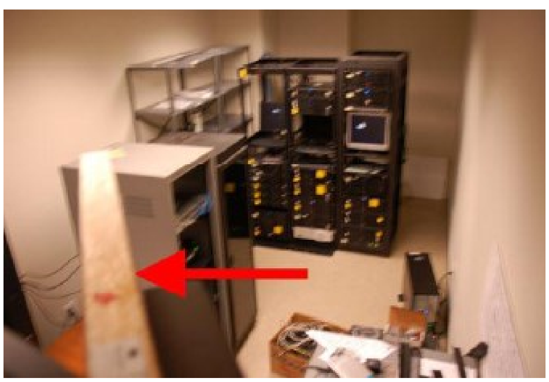


Figure 2-18 Effect of doors on the visibility (CheckVideo, 2014)

Figure 2-19 Full view of objects inside a room (CheckVideo, 2014)

For the purpose of monitoring interior spaces in buildings (i.e. rooms), it is necessary to maximize the view area as much as possible in a way that an individual can fully appear in at least one area, as shown in Figure 2-20. Also, the view should avoid obstructing elements such as furniture, windows and doorways, as shown in Figure 2-21; and avoid targeting external views through windows.

Figure 2-20 Maximizing the view area (CheckVideo, 2014)

Figure 2-21 Obstructing the camera view by the doors (CheckVideo, 2014)

For monitoring building exteriors, it is advisable to place the camera on a fence and not facing directly the building, as shown in Figure 2-22. If this is not possible, the camera has to be offset as much as possible from the building structure and should be horizontally directed. Also, it is important to maximize the view to the entrances of doorways, driveways and pedestrians paths, as shown in Figure 2-23. Finally, it is recommended to reduce the viewing of areas that can cause reflections, such as pavement areas.

Figure 2-22 Placing the camera directly to the wall (CheckVideo, 2014)

Figure 2-23 Maximizing the view outside the building (CheckVideo, 2014)

2.3 Building Information Modelling

2.3.1 BIM concept

According to Eastman et al. (2011), Building Information Modeling (BIM) is a three-dimensional (3D) digital representation of the building process which has the ability to simplify the interoperability and information exchange processes in a digital structure. In other words, BIM technology is providing the opportunity to engage various stakeholders to share information by using one 3D model. The difference between using 2 Dimensional (2D) and BIM technology is that the information of the 2D method is in a graphical format, such as lines, arcs and circles, while in BIM it is represented by specific attributes that describe the building components, such as walls and columns (CRC Construction Innovation, 2007).

Sjogren and Kvarsvik (2007) conducted a comparison between document-centric methods for managing the construction industry in the usage of the BIM method for the same purpose. They outlined, in Figure 2-24, the most critical problems related to the document centric method as follows: (1) the loss of data and the emergence of errors from communication during the same phase of the project; and (2) the repetition of entering data (i.e. around seven times) by diverse systems before submitting the project to the owner.

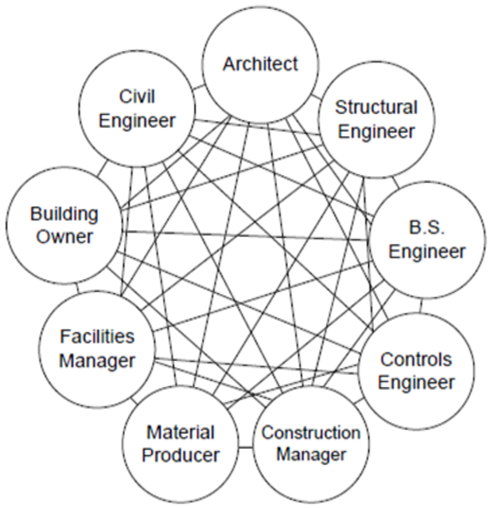


Figure 2-24 Document-centric method (Sjogren and Kvarsvik, 2007)

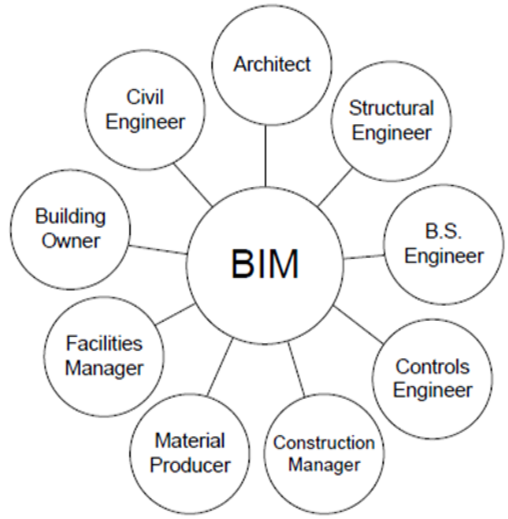


Figure 2-25 Information-centric method (Sjogren and Kvarsvik, 2007)

On the other hand, the Information-centric method, as shown in Figure 2-25, allows all the stakeholders to communicate with each other through a shared language by utilizing the BIM technology and employing a single database system (Sjogren and Kvarsvik, 2007).

According to Barlish and Sullivan (2012), in order to achieve successful results from using BIM, some factors should be taken into consideration as follows: (1) the project size; (2) the proficiencies of the project team; and (3) the efficiency of communication between the team members. Furthermore, the parametric nature of the data which are used in BIM can benefit different project's stages as follows: (1) facilitating the stage of decision making of the design process; (2) establishing quality construction documents; (3) strengthening the performance of assets; (4) providing an accurate estimation of the project cost; and (5) planning the construction processes (Eastman et al., 2011). BIM can also be integrated with other tools in order to achieve different management objectives.

For example, Motamedi and Hammad (2009) explained the building lifecycle management by integrating Radio Frequency Identification (RFID) and BIM.

2.3.2 The role of BIM in the AECOM industry

A study by Gu and London, (2010) defined BIM as an Information Technology (IT) system qualified method that requires utilizing an essential digital representation of all building data throughout various stages of the project lifecycle in a data warehouse. They outlined that BIM is one of the significant domains in the contemporary Virtual Reality (VR) practices, since it can enrich the VR with various graphical and parametric data. VR is just one benefit of what BIM can bring to the architectural, engineering, construction, operation and maintenance (AECOM) industry. In addition, the usage of BIM is affected by many factors, as follows: (1) inefficient training; (2) the complexity of the construction industry; (3) the inability and the lack of encouragement to learn new technologies; and (4) the unclear roles, responsibilities and distribution of advantages.

The study of Asen (2012) clarified that the AECOM industry is a complicated compound that includes different stakeholders, which needs to be coordinated by providing effective communication methods. BIM is offering a solution for this problem, since it has the ability to engage different stakeholders that share the same project in order to participate in achieving the desired outcomes. Asen (2012) mentioned that buildings are becoming more complicated because they contain different and various systems which correlate with each other. The formulation of these correlated systems makes the job harder for stakeholders, especially designers, since they have to take into consideration this complicated relationship in their design analysis. Providentially, the usage of BIM is offering the opportunity to have a database system that includes all building components represented by the form of graphical and non-graphical information. Another benefit of BIM added by Asen (2012) is the ability for different stakeholders to deal with or adjust the status of the building information throughout all stages of the building life cycle. Asen (2012) also evaluated the role of BIM in enhancing the facilities management work. BIM showed high capability in assisting with the maintenance work, as shown in Figure 2-26, by applying a number of visualization methods, namely: (1) icons and symbols; (2) 3D components; and (3) color-coded spaces.

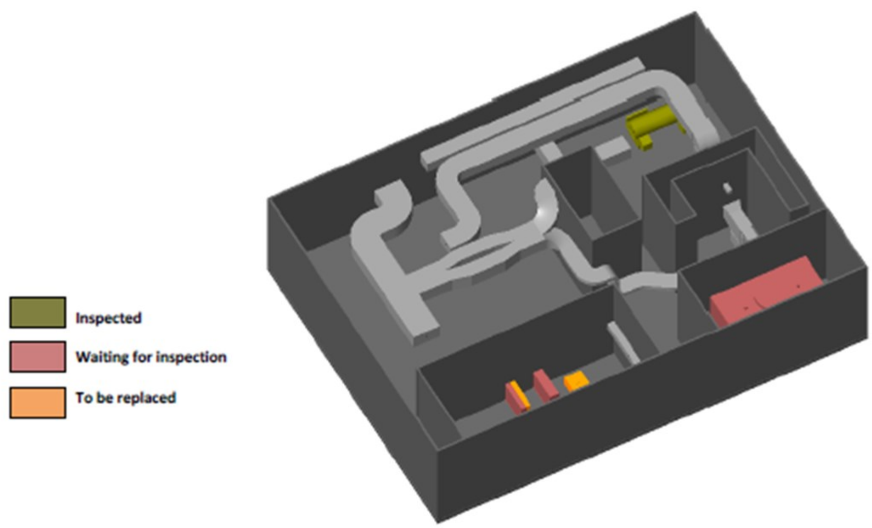


Figure 2-26 BIM role in assisting the maintenance work (Asen, 2012)

2.3.3 BIM and traditional 3D methods

Azhar et al. (2008) conducted a study that focused on the role of BIM in visualizing and simulating the construction projects. The study highlighted the main distinction between BIM and traditional 3D CAD. That study explained that the traditional 3D CAD depends on viewing the building model in separate 3D views such as plans, sections and elevations. The usage of this traditional method is actually prone to errors, since modifying one of these views will require adjusting the others, as shown in Figure 2-27.

In contrast, BIM is more efficient in this case, since the adjustment of any view will be automatically updated to all other views. In addition, BIM is more beneficial than traditional methods, since it provides information not just about graphical entities but also regarding physical and functional data about the building, as shown in Figure 2-28, (Azhar et al., 2008).

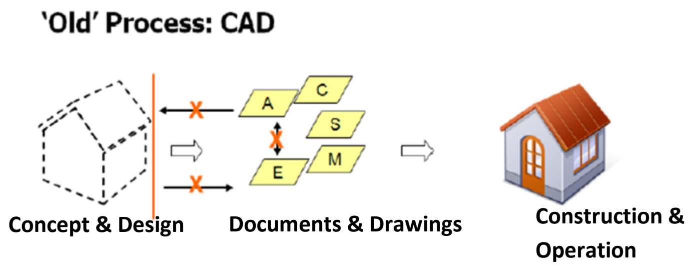


Figure 2-27 Traditional 3D method (Azhar et al., 2008)

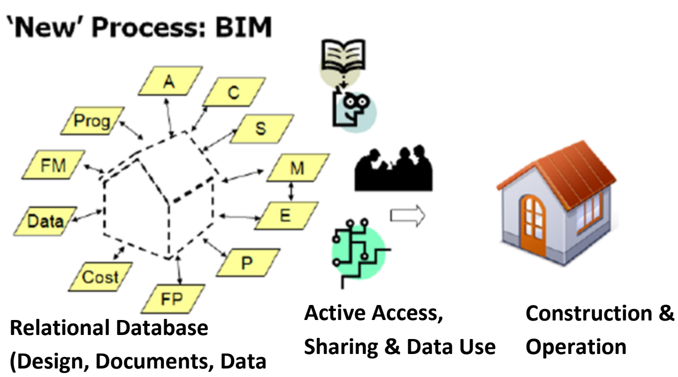


Figure 2-28 BIM method (Azhar et al., 2008)

A thesis of Hergunsel (2011) mentioned that BIM can be used to help the construction manager in producing renderings, walkthroughs, and progression of the model in order to achieve greater communication and understanding of the BIM 3D model. This BIM capability can reduce the conflicts that usually occur between different stakeholders during the construction process. The thesis outlined the role of BIM in providing efficient

4D modeling which has the ability to enhance the understanding of the construction components by using scheduled progress of the construction process, as shown in Figure 2-29. Hergunsel (2011) also highlighted the importance of taking into account the involvement of 4D BIM in the early stages of the construction process. Early involvement will make the project less prone to errors and the outcomes will be achieved as initially planned.

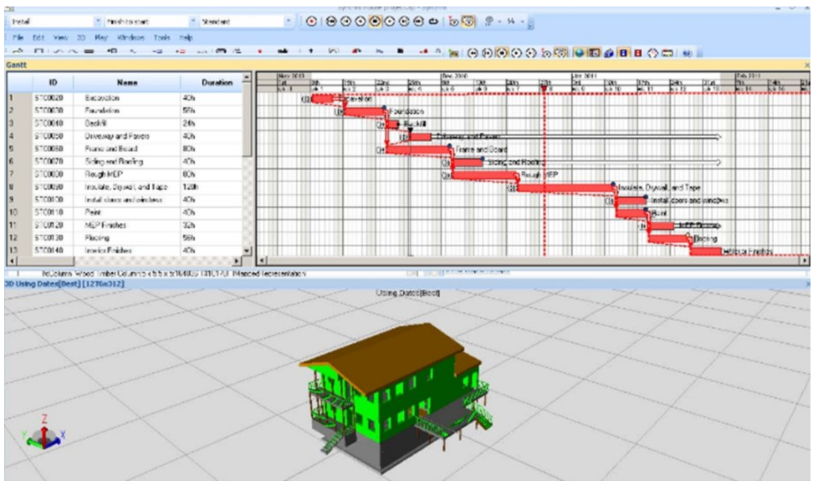


Figure 2-29 Progress schedule of the 4D BIM (Hergunsel, 2011)

2.3.4 Advantages of using BIM

According to CRC Construction Innovation (2007), the key advantage of using BIM is its ability to provide a precise geometrical representation of the building components in a multi-disciplined data environment. Other advantages can be listed as follows:

(1) BIM is offering faster and more efficient processes: data is more simply shared and reused.

(2) BIM can produce a perfect design: building proposals can be carefully checked and simulations can be immediately implemented.

(3) BIM has the ability to make the environmental performance more expected, and the costs of the lifecycle are better understood (e.g. client requirements regarding the environmental performance can be compared as-designed, as-built and as-performing data in the BIM model).

(4) BIM can provide greater customer service by using precise visualization: proposals can be much better explained.

(5) BIM is able to integrate lifecycle data: this data can be related to requirements, design, and construction, and operational data can be used in the field of facilities management. That means, BIM model is able to hold various information in a single data system considering the accuracy and the consistency of data.

2.3.5 BIM and security

BIM can facilitate the process of integrating the security system with other operating systems, since it permits the contractor to validate the system design and allows the owner to achieve successful decisions (Bremer, 2011). Furthermore, Axis communications (2013) and BOSH (2015) developed different types of 3D camera models in order to be used in Autodesk Revit (2014). These models allow designers to select different camera features and to visualize different security issues inside BIM model. 3D camera models can facilitate the design process, and can visually detect constraints, which are not observable when using 2D drawings (Bernard, 2014). However, checking the camera coverage using 3D visualization does not give accurate coverage results, especially in places of blind spots behind obstacles. Also, finding the suitable placement of these 3D camera models is not an easy task, due to the huge number of configurations and constraints, which affect the placement.

2.3.6 Challenges of implementing BIM

Although the usage of BIM showed a clear impact on achieving projects' objectives, this technology is still facing some challenges related to cost, information management and the need to improve the visualization feature of this technology. BIMhub (2013) listed some of the challenges as follows:

(1) The lack of expertise: since this technology is new on the market, many companies complained about the complexity of using BIM and the need to have more expertise in this technology.

(2) The reluctance issue: the reluctance to change and to learn new technologies, especially skilled and expert individuals who prefer to execute their tasks using their own ways.

(3) Data management: since BIM is a technology that requires interoperability and gathering different information from different sectors, this makes the process of managing information much harder for some companies.

(4) The cost of the technology: some companies believe that adopting BIM software is more costly than traditional methods, since it will require the usage of new computer systems, in addition to the need to employ skilled staffs who have knowledge of the 3D modeling process.

2.4 Optimization

A simple definition for optimization is: the mathematical algorithm that can find the best solution for a mathematical problem which is restricted to a number of ranges and/or options. According to a research by Mawlana (2015), optimization can be categorized as follows: (1) the number of objective functions; (2) the certainty in the decision variables or the values of objective functions; and (3) the optimization problem type. The research outlined that there are two types of optimization problems. The first type is named a “single objective problem” because it has a single objective. The second type is named a “multiple objective problem” because it has multiple objectives. Also, the optimization is affected by the value of the objective function. If the objective function value of the decision variables can be estimated with certainty, then the optimization is considered as deterministic. In the case where the objective function value cannot be estimated with certainty, then the optimization is considered as stochastic.

2.4.1 Optimization types

Determining the type of optimization problem is dependent on the form of the equations that formulate the objective functions and the selected decision variables. According to Diwekar (2008), these equations can be linear, nonlinear, or discrete. Optimization of the problems that have linear objective functions and constraints, and scalar and continuous decision variables, is called linear optimization, while optimization of problems that have nonlinear objective functions and constraints, or either of them, is named “nonlinear optimization”.

Mawlana (2015) added that linear and nonlinear optimization should have continues decision variables. Furthermore, discrete optimization is another type of optimization which usually has discrete decision variables. There are three types of discrete optimization as follows: (1) integer programming; (2) mixed integer linear programming; and (3) mixed integer nonlinear programming. The integer programming type involves scalar and integer decision variables, and can include linear or nonlinear constraints. The mixed integer type is integration between integers and continuous decision variables. In other words, mixed integer linear programming includes linear objective functions and constraints, while mixed integer nonlinear programming includes nonlinear objective functions and constraints.

2.4.2 Multi-objective optimization

Abraham et al. (2005) mentioned that the search space is usually clear in single objective optimization, while there is more than one optimal solution in the case of multi-objective optimization, which is usually related to multiple objectives and where the search space is not clear. According to Cohon (1978), multi-objective optimization is a process that aims to determine a variety of effective solutions to a problem by utilizing a number of steps. The purpose of using mathematical functions is to establish the objective functions and constraints in order to specify the decision variables (Nakayama et al., 2009). As a result, according to Abraham et al. (2005), multiple objective optimization problems can be mathematically formulated as follows:

$$minimize\ [f_1(x), f_2(x), .., f_k(x)] \qquad \text{Equation 2-1}$$

Subject to the m inequality constraints:

$$g_i(x) \leq 0 \quad i = 1,2, \dots., m \qquad \text{Equation 2-2}$$

and the p equality constraints:

$$h_i(x) = 0 \quad i = 1,2, \dots., p \qquad \text{Equation 2-3}$$

Where k is the number of objective functions *f(x),* $x = [x_1, x_2, \dots., x_n]$ is the vector of the decision variables, *f(x)* is the set of objectives to be minimized, $f_1(x)$ is the function of the first objective, *g(x)* and *h(x)* are the functions of the sets of inequality and equality constraints.

A multi-objective optimization problem is associated with a number of objective functions that can be conflicting, non-conflicting, or partially conflicting (Goh and Tan, 2009). The complicated nature of the multi-objective optimization problems is a result of their multiple, contradictory objectives with an outsized search space (Jimenez, 2007). Practically, most problems have contradicting objectives, which make the process of determining a single optimal solution difficult to achieve. According to Jimenez (2007), two methods are used to solve the multi-objective optimization problem: (1) integrating the multiple objective functions into a single objective function; or (2) having a set of non-dominated optimal solutions (Jimenez, 2007). The final result is a group of optimum solutions which represent the trade-off process with their related objectives.

The concept of trade-off is an improvement process of an objective function by the deterioration of another objective function. Abraham et al. (2005) declared that in cases of solving multi-objective functions, it is preferable to look at the trade-off solutions instead of the single solutions. Each solution resulting from the optimization process is considered as a Pareto solution or non-dominated, which is based on the concept of trade-off. The final optimal solutions are called the Pareto set or the non-dominated set. Furthermore, for the purpose of solving the trade-off problem, two mathematical techniques are used as follows: (1) mathematical programming; and (2) metaheuristic search methods. The mathematical programming methods, such as linear programming and integer programming, represent the objective functions and constraints as a closed-form formula (Mawlana, 2015). These mathematical programming methods showed some limitations, such as: (1) the process of formulating the constraints and the objective functions required too much time to be conducted (Liu et al., 1995); (2) optimum results are not guaranteed by some of these methods (Feng et al., 2000); (3) cannot deal with multiple objective functions (Zheng et al., 2005; Reddy and Kumar, 2007); and (4) they are not efficient in evaluating large complex problems (Adeli and Karim, 1997; Senouci and El-Rayes, 2009). Metaheuristic methods are a way to solve contradictory problems that have non-deterministic polynomial time (NP). NP problems are the hard problems of mathematics where the solving time cannot be simply specified because the number of solutions exponentially increases when the camera configurations are increasing (Murray et al., 2007). Metaheuristic methods can be categorized into: (1) population-based methods, such as Particle Swarm Optimization and Genetic Algorithms, and (2) trajectory methods, such as Tabu Search and Simulated Annealing (Mawlana, 2015).

2.4.3 Genetic algorithms

The Genetic Algorithm (GA) is one of the well-known metaheuristic methods that can solve different NP hard problems (Mawlana, 2015). According to Abraham et al. (2005), GA is also named as "evolutionary optimization", which has the ability to generate a population of candidate solutions. These candidate solutions will be integrated with each other in order to produce new solutions, as shown in Figure 2-30.

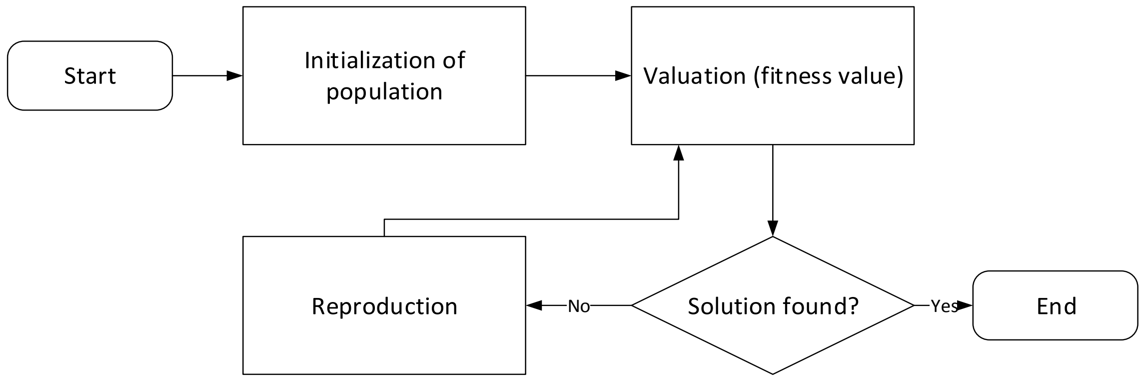


Figure 2-30 Evolutionary optimization process (Abraham et al., 2005)

The GA method gives the ability to select several numbers of a Pareto-optimal set in one running cycle of the optimization, rather than conducting a sequence of them. This can be achieved by searching diverse portions of the solution landscape in a parallel method (Yu and Gen, 2010). Abraham et al. (2005) also highlighted the comparison between GA and other metaheuristic optimization methods. According to Mawalana, (2015), GA can offer the opportunity to specify the global near-optimal trade-offs, and it does not stop at a local optimum like the other metaheuristic methods. GA is considered more beneficial than other methods because of the following: (1) the ease of solving the multi-objective optimization problems; (2) it can be applied to multi-objective optimization problems even in cases where the problem representation is complicated (Deb, 2001), (3) it is not affected by the Pareto-front shape (Goh and Tan, 2009), and (4) the ease of applying GA in parallel environments (Abraham et al., 2005).

2.5 Research related to camera placement

Many works have been conducted to enhance the performance of surveillance cameras by addressing a number of issues associated with these cameras. Table 2-3 shows a summary of some of these works.

Table 2-3 Summary of some previous works related to surveillance cameras

Work	Description
Lee et al. (1995)	Evaluating a number of guidelines of mounting Surveillance cameras.
Bigdelil et al. (2007)	Evaluating the capability of security camera systems on public railway platforms.
Chae and Kano, (2007); Bohn and Teizer, (2009); Yang et al., (2010); and Starzyk and Qureshi, (2011)	Evaluating the capability of surveillance cameras in facilitating tracking and monitoring purposes.
Tomioka et al., (2010, 2011); and Ogino et al., (2014)	Finding the optimum placement for mobile surveillance cameras using mixed integer linear programming.
O'Rourke (1987)	Proposing the theory of Art Gallery Problem.
Murray et al., (2007) Kim et al. (2008) and Janos et al. (2007)	Optimizing the camera placement in a specific location of an urban area.
Yabuta and Kitazawa (2008)	Optimizing the camera placement by using a 2D method.
Nam and Hong (2012)	Optimizing camera number, type and placement using 2D method.
Chen et al. (2013)	Evaluating the role of BIM technology in determining the suitable tilt angle of the camera.
Sinha et al. (2006) and Dornaika and Elder (2012)	Proposing a network of various types of cameras to build a panoramic view.
Erdem and Sclaroff (2006)	Finding the optimum PTZ camera placement for a given area.
Zhao and Nevatia, (2004); Qureshi and Terzopoulos, (2006); Tu et al., (2007); and Starzyk and Qureshi, (2011)	Improving the tracking, resolution and face detection tasks of the PTZ camera.
Qureshi and Terzopoulos, (2009)	Optimizing the planning for PTZ camera tasks.
Toregas et al. (1971)	Proposing the theory of the location set coverage problem.
Lee et al. (2012)	Proposing a method of multi-objective optimization approach for sensor placement.
Amriki and Atrey (2012)	Optimizing the camera placement inside a bus using 3D method.

Lee et al. (1995) conducted a study that aimed to evaluate a number of guidelines established by the California Department of Transportation (1994) for siting and mounting surveillance cameras. The evaluation depends on the design experience of locating 100 surveillance cameras on the freeways and streets of California. Some of the evaluated guidelines can be listed as follows: (1) the importance of increasing the coverage for areas close to ramps and intersections; (2) ensuring an adequate view of accidents by using historical data; (3) ensuring appropriate maintenance access through the area design; (4) decreasing the view of physical obstacles; and (5) taking into consideration different related services such as underground utilities. The study concluded that once the surveillance cameras are placed on a site, the process of changing their location is too costly. Bigdelil et al. (2007) evaluated the capability of security camera systems applied on public railway platforms. The study aimed to achieve robust detection and identification of individuals in crowded situations. They proposed a configuration real-life trial system, as shown in Figure 2-31, that includes various video analytic systems which are used to detect an event of interest. The system was composed of existing analogue cameras, video switch, digital video recorder, video analytic software, 3D immersive video presentation layer, and a 3D model of the railway platform. The study found a number of limitations of this system, such as low resolution and few frames recording per second. In addition, they outlined that privacy laws were obstacles during their research. That is because most privacy groups were not comfortable with this security research. They also found that it was too time consuming to implement the process of setting up commercial systems for diverse numbers and types of cameras. This is because every camera has different capabilities, provided by different commercial software companies. In order to avoid the previous limitations, it is very necessary to evaluate the cameras' capabilities and their surrounding conditions during the design phase and before completing the construction process.

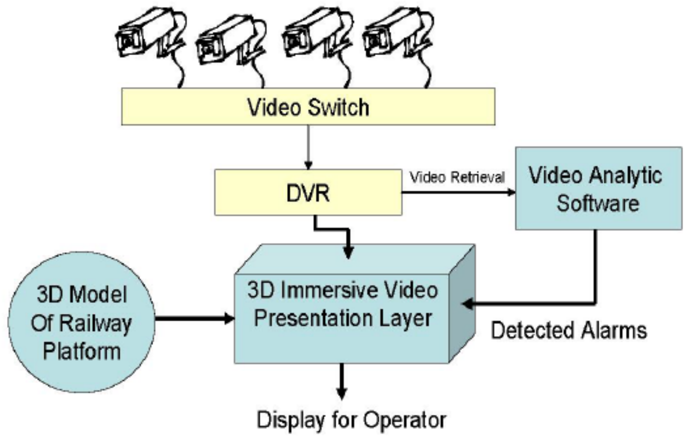


Figure 2-31 Real-life trial system (Bigdelil et al., 2007)

Some researches (Chae and Kano, 2007; Bohn and Teizer, 2009; Yang et al., 2010; and Starzyk and Qureshi, 2011) evaluated the capability of surveillance cameras in facilitating tracking and monitoring purposes. For example, the study of Starzyk and Qureshi (2011) developed a behaviour-based controller for passive and active cameras for the purpose of increasing the ability of surveillance cameras to handle multiple scenes at one time. They proposed a system composed of a number of cameras that can adjust their parameters based on the complexity of the scene. Although it seems to be ideal to capture multiple scenes simultaneously, their method lacks the consideration of important factors that can certainly affect their system such as: (1) the existence of different types of cameras and their features (e.g. FOV); (2) the existence of physical obstacles that can affect the camera visibility; and (3) the impact of the camera position on the system performance. Other studies (Tomioka et al., 2010, 2011; Ogino et al., 2014) sought to find the optimum placement for mobile surveillance cameras using mixed integer linear programming.

A number of studies were implemented to address these issues. For example, a study by O'Rourke (1987) proposed a theory named "the Art Gallery Problem", which states the possibility of covering a specific location that has a polygon shape by allocating a number of guards in the corners of this shape. This theory states that the more guards added, the more opportunity there is to fully cover the polygon shape. Even though this theory seems to be efficient as regards the camera placement issue, it includes two major limitations. First, the assumption that adding more guards will increase the visibility is not accurate, since it does not consider the limitation of the guard's ability (or in the camera placement issue, the limit of the FOV). Instead, it was assumed that the guard has unlimited visibility. Second, the assumption that the area always has a polygon shape is also not accurate. As a result, too many guards would be required to cover areas of a more complex shape (Murray et al., 2007). A study by Toregas et al. (1971) proposed a theory named the "location set coverage problem". This theory aims to find the optimum placement for a number of facilities (i.e. the minimum number of facilities) to fulfill the demands of an area, through fixing a specific distance that the facilities can reach. The limitation of this theory can be represented by the fact that fixing the distance might result in the need to employ too many facilities that cannot be afforded by the supplier.

2.5.1 Surveillance camera placement using 2D and 3D methods

Yabuta and Kitazawa (2008) proposed an algorithm for optimizing the camera placement by using a 2D method for calculating the camera coverage. The study solved the camera placement problem by taking into account a number of specifications, namely, the visual distance, the visual angle and the resolution. First, their method included dividing the targeted area into rectangular regions called segments. The segmentation process, as shown in Figure 2-32, begins with locating the data of the targeted area (i.e. area data and block data) into a 2D plan. Then the study proposes extending marginal lines from the rectangular shape of the block areas and merging insignificant regions that shared the same end line, and which were not attached to a block corner. The second step of the method is implementing a visibility test, which mainly depends on connecting the line segments to the center of regions in order to obtain the required visibility. The final steps include setting a covering problem formulation and solving this problem formulation respectively, creating the problem simulation, achieved through adopting two algorithms, one named "the all regions observation" and the other named "the waited regions observation".

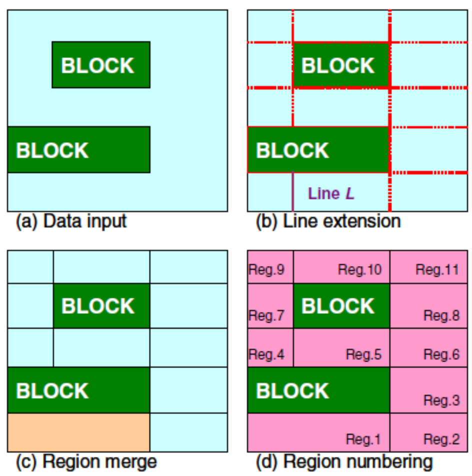


Figure 2-32 Proposed segmentation process (Yabuta and Kitazawa, 2008)

The study found that by using the waited regions observation algorithm, the number of cameras required was decreased by approximately 50%; and that using multiple cameras would help to decrease the number of cameras required and increase the visibility regions. The study clarifies that the most efficient algorithm uses the multiple camera types with the waited regions observation. However, the study has a number of limitations as follows: (1) the study mainly depended on the implementation of 2D assumptions, which is inadequate to cover some areas hidden behind building objects or even located down from the FOV of the camera, which might affect the visibility, as discussed in (Albahri and Hammad, 2015); (2) the adopted segmentation process is not accurate, since it is representing different sizes of areas. This will make the visibility focus on covering the area as a whole (since the size of the segment is taking the size of the area), and not detect objects and movement inside these areas. Many practices implemented different techniques to decrease the impact of these issues.

On the other hand, other practices, such the study of Nam and Hong (2012), aimed to find a method for optimizing camera number, type and placement. The study sought to optimize the placement of multiple cameras in indoor spaces, taking into consideration the coverage and the cost constraints. Specifically, the study proposed a scientific method of optimal camera placement, which includes four steps: space modeling, agent modeling, generating trajectories and selection of optimal camera placement. The study also suggested a developed path-finding algorithm in order to determine the actual path travelled by people. The study proposed a simulation tool that included two types of data: (1) the area of interest, and (2) the camera type. The study found that in order to gain precise measurements in the simulation process, it is necessary to take into consideration the computational cost of the cameras used and the cost of the installation process. However, this study lacked consideration of a clear method of surface segmentation of the targeted area, which can help in accurately determining the locations of the important areas and obstacles. Also, the process of imitating the movement of people for the purpose of specifying the important areas does not always give accurate results, due to the fact that sometimes there are places that do not allow the movement of people or do not permit them to be covered by surveillance cameras. This problem can be fixed by adopting the BIM technology which has the ability to identify location identities (i.e. important and unimportant locations) with far more accurate results than other methods. Morsly et al. (2012) used the method of Binary Particle Swarm Optimization Inspired Probability (BPSO-IP) to find the optimum camera placement. They used a grid system to facilitate the camera placement on a discrete grid points. Each camera placement is represented by a bit of X, Y, and the horizontal angle. Their experiment shows successful results of automatic camera placement.

Murray et al. (2007), Kim et al. (2008) and Janos et al. (2007) implemented a coverage optimization process in a specific location of an urban area. They aimed to maximize the surveillance coverage in an urban area by achieving efficient camera deployment and taking into account cost constraints. Also, they conducted a visibility analysis, called the line of sight, to calculate the coverage of camera locations, as shown in Figure 2-33.

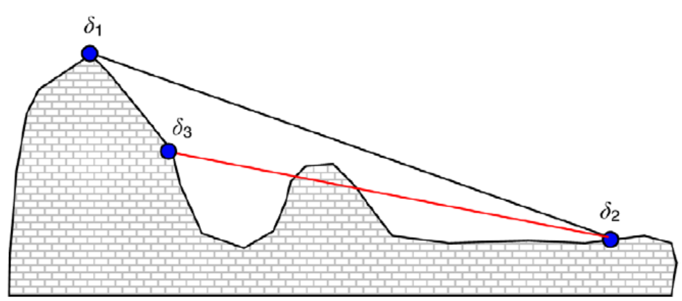


Figure 2-33 Visibility analysis (Murray et al., 2007)

Two optimization techniques were applied, one called “the Maximal Coverage Location Problem” (MCLP) and the other called “the Backup Coverage Location Problem (BCLP)”, in order to achieve optimal placement of multiple cameras. To achieve their objectives, they used a linear optimization method to solve these two problems. The MCLP method has the ability to maximize the coverage area with the least number of cameras. Figure 2-34 showed a trade-off between the number of placed cameras and the coverage amount. The curve showed a proportional increase in the coverage by adding more cameras. The BCLP method focuses on maximizing the overlapping coverage for a number of cameras. Their methods are formulated by the following equations:

(1) To solve the MCLP problem:

$$\text{Maximize } Z = \sum a_j \, y_j$$ Equation 2-4

a_j: is the importance of area j $y_j = 1$ if area j is covered and $y_j = 0$ otherwise

j: index of demand areas i: index of potential camera placement

$$\text{Subject to } \sum_{i \in Nj} x_i - y_j \geq 0 \qquad \forall j$$ Equation 2-5

x_i=1 if potential camera i is located and $x_i = 0$ otherwise

$$Nj = \{i | \boldsymbol{\lambda} = \mathbf{1}\}$$ Equation 2-6

$$\sum_i x_i = p$$ Equation 2-7

where p is the number of sensors to cover

$$x_i = (0,1) \quad \forall i$$ Equation 2-8

$$y_j = (0,1) \quad \forall j$$

(2) To solve the BCLP problem:

Objective 1: $$Maximize\ Z_1 = \sum_j a_j y_j$$ Equation 2-9

Objective 2: $$Maximize\ Z_2 = \sum_j a_j u_j$$ Equation 2-10

Where u_j=1 if area j is covered more than once, while u_j= 0 otherwise.

$$Subject\ to \sum_{i \in Nj} x_i - y_j - u_j \geq 0 \quad \forall j$$ Equation 2-11

$$u_j - y_j \leq 0 \quad \forall j$$ Equation 2-12

$$\sum_i x_i = p$$ Equation 2-13

$$x_i = (0,1) \quad \forall i$$ Equation 2-14

$$y_j, u_j = (0,1) \quad \forall j$$

Also, the two objectives can be written as the following:

$$Maximize\ Z = w \sum_j a_j y_j + (1-w) \sum_j a_j u_j$$ Equation 2-15

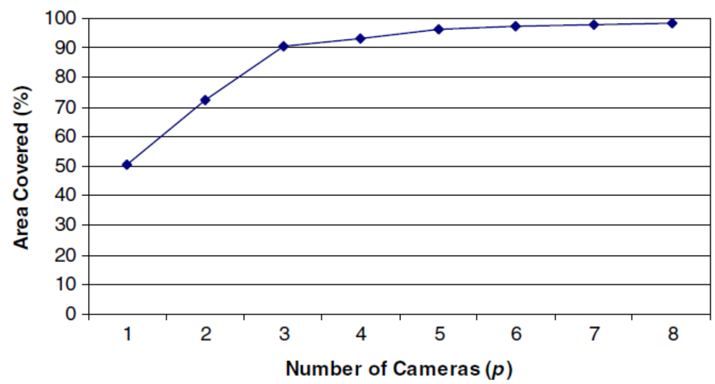


Figure 2-34 Area covered by a number of cameras using MCLP (Murray et al., 2007)

Their method showed high capability of tracking the movement of people and activities. The limitations of this work can be outlined as follows: (1) their method was implemented in external urban locations where the search space is limited in comparison with the search space of indoor building spaces; (2) in their visibility analysis they divided the study area into cells; these cells were formulated using a 2D grid system which may affect the visibility, as previously discussed, since it does not represent the real volume of the static and movement objects; (3) because the work was implemented outside buildings, the method does not consider the existence of different degrees of important areas which usually exist inside buildings; and (4) finally, based on the authors using a GIS to calculate the coverage, it is not an exact 3D method, instead, it is a 2.5D method, which cannot deal with complex shapes such as arches and tunnels.

Lee et al. (2012) proposed a method of multi-objective optimization approach for sensor arrangement. The method showed high capability in optimizing the placement of a number of Radio Frequency Identification (RFID) sensors. Their experiments showed successful multiple Pareto solutions which were simulated on a number of 2D maps. However, the placement RFID sensors using 2D method is not always helping the placement problem of surveillance cameras. This is because of the lack of considering the height which can affect the visibility of some hidden spots behind obstacles inside buildings. Also, the assumption that the targeted area is equal important will requires covering the whole area by adding more sensors which is inefficient.

The study of Amriki and Atrey (2012) used a 3D optimization method inside a bus. They aimed to investigate a scientific method that can find a suitable camera placement by specifying the number of required cameras and their placement. In order to address the placement issue, the study proposed the development of a 3D bus model in order to help in the process of detecting the covered and uncovered areas inside the bus. This model uses equal cubic cells. The size of these cells is as small as the smallest space between the bus seats. Each cell is assigned with a 3D location, an importance degree and a grouping value. This study calculated the coverage by using the following equations:

$$W\Phi = \sum_{i=1}^{n} C_i I_i \qquad \forall\, C_i \in \Phi$$

Equation 2-16

Where Φ the array of the cells, C_i is the number of covered cells, I_i is the importance value of covered cells and W is the weight of this array of cells.

$$W\ segment = \sum_{i=1}^{n} C_i I_i \qquad \forall\, C_i \in \Phi\, Segment$$

Equation 2-17

$$\Phi_k \subseteq \Phi$$

$$G_k = \frac{\sum_{i=1}^{n} C_i \, I_i}{W\Phi} \quad \forall \, C_i \in \Phi_k \qquad \text{Equation 2-18}$$

Where G_k is the gain (percentage) of coverage inside the bus by a camera k.

$$G_k segments = \frac{\sum_{i=1}^{n} C_i \, I_i}{W\Phi \text{ segments}} \quad \forall \, C_i \in (\Phi\text{segment } \cap \Phi_k) \qquad \text{Equation 2-19}$$

The study also minimizes the number of camera configurations by utilizing a binary optimization method called "the SmartMax algorithm". Amriki and Atrey (2012) highlighted that by reducing the cell's size, the number of cells will increase and the computational time will increase but the result will be more accurate. Applying the SmartMax algorithm has the purpose of reducing the coverage computation time. SmartMax mainly focuses on finding the primary coverage with the least number of camera configurations. Also, the study included a segmentation process which divided the bus into three types: a door segment, a seats segment and a pathway segment, as shown in Figure 2-35.

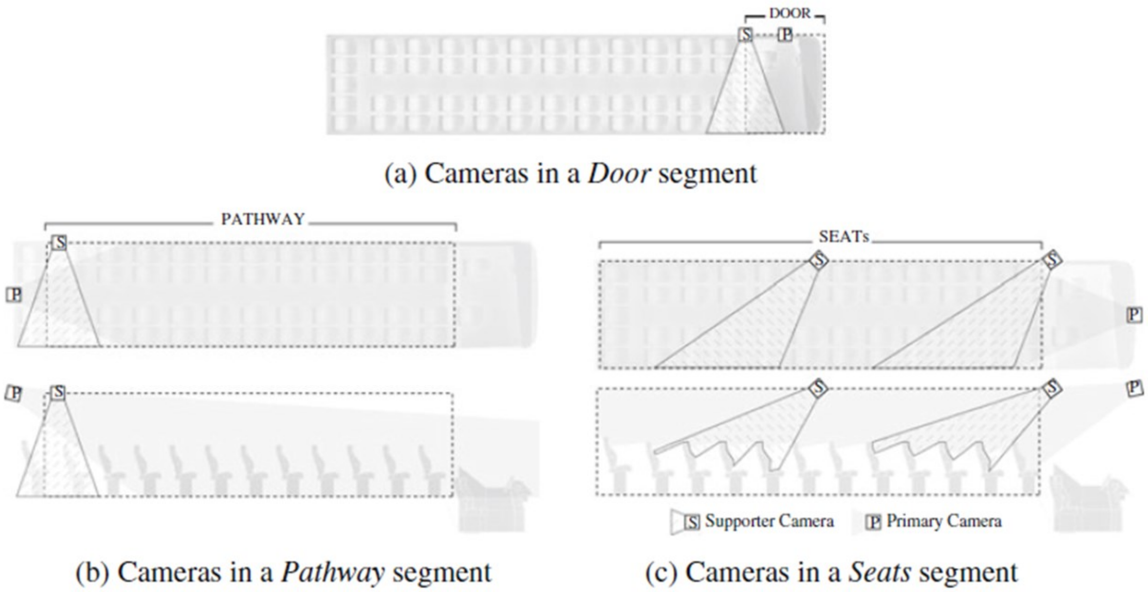


Figure 2-35 Segmentation process inside the bus (Amriki and Atrey, 2012)

This study showed rational placement of cameras as well as a relative increase in coverage based on the number of cameras. However, considering the more complex indoor building spaces, this method lacks the geometrical constraints, interrelated systems and interior classification of spaces inside buildings. The camera can be placed at any location without any consideration given to the existence of heat and air conditioning sources or other systems. Also, their optimization method did not take into account the possibility of using different types of cameras (e.g. cameras with different FOVs), which is the real case when placing cameras inside buildings.

2.5.2 BIM role in camera placement

According to a report by OPTICOM (2013), vibration is the main factor that can affect the life expectancy of surveillance cameras. Therefore, proper design of the camera placement is necessary to avoid the effects of vibration. The report highlighted the importance of placing the surveillance cameras in suitable positions that can protect them from vibration and, at the same time, allows them to provide adequate views. Examples of vibration sources are HVAC systems which usually are placed on the ceiling. Chen et al. (2013), outlined that the existence of smoke curtains can affect the performance of the cameras.

According to Moog Videolarm (2012), heat sources (e.g. heaters) are factors that may cause the failure of surveillance cameras. As a result, installers prefer to change the height of the cameras in order to avoid these elements. Consequently, changing the height of cameras can affect their coverages. Therefore, the location of these sources should be considered before placing surveillance cameras. Chen et al. (2013) implemented a study that aimed to evaluate the role of BIM technology in determining if the design of the surveillance cameras is suitable to meet the future security system requirements when the construction process is completed. The work defines a number of significant parameters which can help in the process of setting up camera location, such as eye elevation and target elevation.

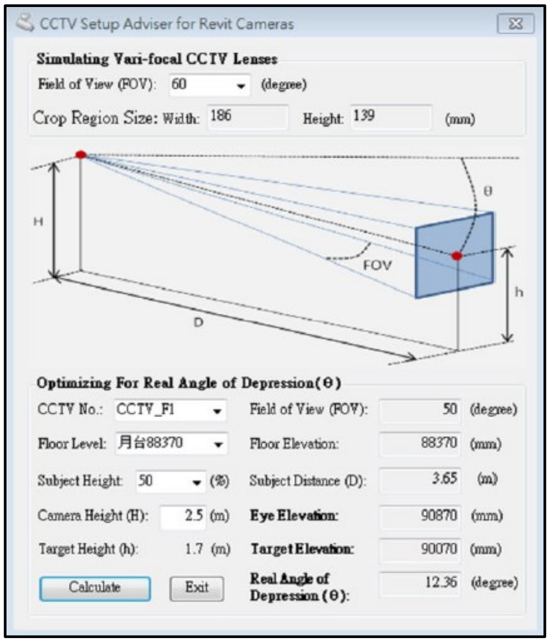


Figure 2-36 Proposed API plug-in (Chen et al., 2013)

These parameters were used for the purpose of achieving the real angle of depression, which can be determined by applying the following equation:

$$\tan\theta = \frac{H - h}{D} \qquad \text{Equation 2-20}$$

Where H is the camera height, h is the target height and D is the distance between the camera and the target.

The work also established an Application Programming Interface (API) in Autodesk Revit, as shown in Figure 2-36, which is used to calculate the real angle of depression considering previous parameters. This API can help in accurately simulating varifocal lenses and CCTV camera views.

The work highlighted the importance of this plug-in in defining the design requirements of the surveillance camera design system, such as the ability to ensure the appropriate coverage in places like main entrances. The work indicated the benefit of BIM model in finding physical obstacles that may affect the camera coverage in a specific location, by using the clash detection method during the design phase. In addition, the work included modeling 3D smoke curtains and ceiling signboards as a way to examine the effect of these 3D models on the coverage of the surveillance cameras, as shown in Figure 2-37.

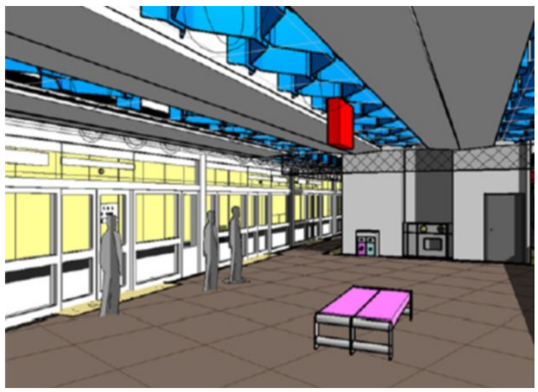


Figure 2-37 Designing Curtains and Signboards Using BIM model (Chen et al., 2013)

They also utilized an efficient 2D measurement tool named "the Fill Region Tool", which helped in automating the calculation process of the coverage areas. In fact, this work strongly supports the claim of this research regarding the efficiency of using the BIM technology in the camera placement issue. However, this work did not take into consideration the process of optimizing camera selection, which should include camera number, type and placement. Instead, they focused on optimizing the real angle of depression of a number of placed cameras. Also, the study calculated camera coverage by using the Fill Region Tool, as shown in Figure 2-38, which has many limitations, due to the usage of the 2D calculation method, as previously discussed. Other research (Cowan and Kovesi, 1988; Williams and Lee, 2006; Bigdeli et al., 2007; Bodor et al., 2007; Becker et al., 2009; Fehr et al., 2009; Gonzalez-Barbosa et al., 2009 and Yao et al., 2010) conducted 2D and 3D techniques for the camera placement issue but without considering the effect of different geometrical constraints on the camera coverage inside buildings. Table 2-4 summarizes some of the most relevant papers related to the current research objectives.

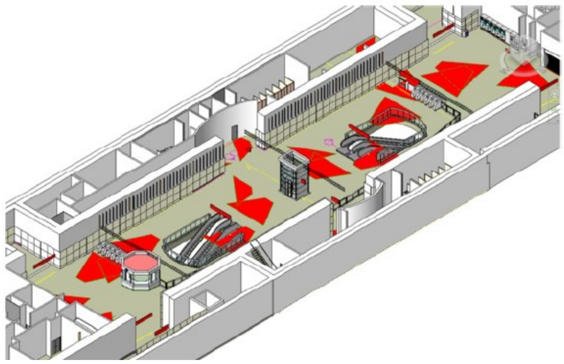


Figure 2-38 CCTV Coverage Using Fill Region Tool (Chen et al., 2013)

2.5.3 Research related to PTZ camera

Recent papers associated with PTZ cameras were focused on three lines of research as follows: (1) enhancing the detecting, tracking and recognition abilities of the PTZ cameras; (2) optimizing the handoff planning between PTZ cameras; and (3) optimizing the placement of PTZ cameras using 2D methods. A number of studies focused on enhancing the image processing of the camera by considering the following factors: (1) resolution; (2) view angle; (3) visibility and (4) privacy areas (Spletzer and Taylor, 2001; Yi et al., 1995; Maver and Bajcsy, 1993; Reed and Allen, 2000; and Tarabanis et al., 1996).

A number of papers focused on enhancing the image processing of PTZ cameras. Sinha et al. (2006) and Dornaika and Elder (2012) proposed a network of various types of cameras including PTZ cameras to build a panoramic view using images with various levels of resolution. Detailed information was extracted by a handoff planning process between the installed cameras. Another research direction focussed on improving the handoff planning between PTZ cameras. Lalonde et al. (2007) proposed a system of PTZ cameras to track pedestrians and vehicles, considering the zooming feature. Although their system showed good results of tracking single activities, it does not consider the suitable distance between the cameras and the targets. Qureshi and Terzopoulos (2007 and 2009) and Starzyk and Qureshi (2011) aimed to optimize the handoff planning between PTZ cameras. Their studies proposed a planning strategy for managing a network of PTZ cameras and evaluating a number of tasks of automatically capturing videos of individuals in a designated area. Although the method proved the feasibility of the PTZ planning process, the method is implemented in a tight area (i.e. a corridor), which is not always the case (e.g. when placing the cameras in large areas like halls). Also, this research used a 2D method, which has limitations regarding the detection of hidden spots behind obstacles. Another line of research is to optimize the placement of PTZ cameras. Erdem and Sclaroff (2006) proposed a method to find the optimum PTZ camera placement for a given area. The objective was to find the optimum placement with the least number of PTZ cameras and considering a number of task-specific constraints. They represented the monitored areas to be covered as 2D grids. However, they implemented their optimization method in a 2D environment without considering the effect of the camera tilt angle, which can affect the coverage results.

2.6 Summary

This chapter implemented a comprehensive literature review for the purpose of determining the elements and standards the can affect the issue of the camera placement. The literature review included the following: (1) reviewing the configurations, parameters and features that affect the selection and placement of cameras; (2) reviewing the rules that that should be considered during the camera placement process; (3) reviewing the role of BIM in addressing several issues close to the camera placement issue; (4) reviewing the existing optimization methods that can assist the research methodology; and (5) reviewing the related practices that focus on the camera placement issue.

Table 2-4 Summary of the most related research

Paper	Method	Objective	Limitations
Amriki and Atrey(2012)	3D	Optimizing the placement of multiple cameras inside a bus considering cost constraints.	• Lack of considering the placed elements on the ceiling due to the difference between the constraints inside the bus and the constraints inside the building.
Chen et al. (2013)	BIM	Optimizing the real angle of depression to achieve optimum view of the placed surveillance cameras.	• This work did not take into account the process of optimizing the camera selection which should include cameras' number, types and placement.
Murray et al. (2007)	3D	Finding the optimum number, type and placement of surveillance cameras in an urban area.	• The search space is limited in comparison with indoor building spaces. • Inefficient visibility methods because of using 2D grids to create cells on the monitored area. • Not considering the existence of different important areas. • Using GIS in the visibility analysis cannot detect complex 3D shapes.
Nam and Hong (2012)	2D	Optimizing the placement of multiple types of cameras taking into account the cost constraints.	• Not considering a clear method of surface segmentation of the targeted area • Simulating the people movement to specify the important areas is not always giving accurate results. • The 2D method is inadequate to calculate the coverage.
Yabuta and Kitazawa (2008)	2D	Optimizing the placement of multiple cameras.	• The 2D method is inadequate to calculate the coverage. • The segmentation process is not accurate; because of irregular sizes of the segments.

CHAPTER 3: RESEARCH METHODOLOGY

3.1 Introduction

This chapter presents the research methodology for single and multiple camera coverage calculation and a method to optimize the placement, number and types of cameras inside buildings. The methodology also includes a part that focuses on optimizing the placement of PTZ cameras inside buildings. The input to this work is a BIM model which includes building and camera information that can affect the camera coverage. The core of this work is a BIM-based method using GA optimization to find the optimum camera type, number and placement. The output of this methodology will be 3D layouts of near-optimum camera placement inside the BIM model. These layouts are representing alternatives of the near-optimum camera placements considering maximum coverage and minimum cost. The suitable layout is the one that can meet the users' requirements.

3.2 Research methodology

After conducting a comprehensive literature review associated to this research, the research proposes three methods, as shown in Figure 3-1, that are necessary to achieve the research objectives: (1) developing a method for calculating the camera coverage; (2) developing a method for optimizing the fixed camera placement inside buildings; and (3) developing a method for optimizing the placement of a PTZ camera inside buildings. Each of these methods will be explained in this chapter.

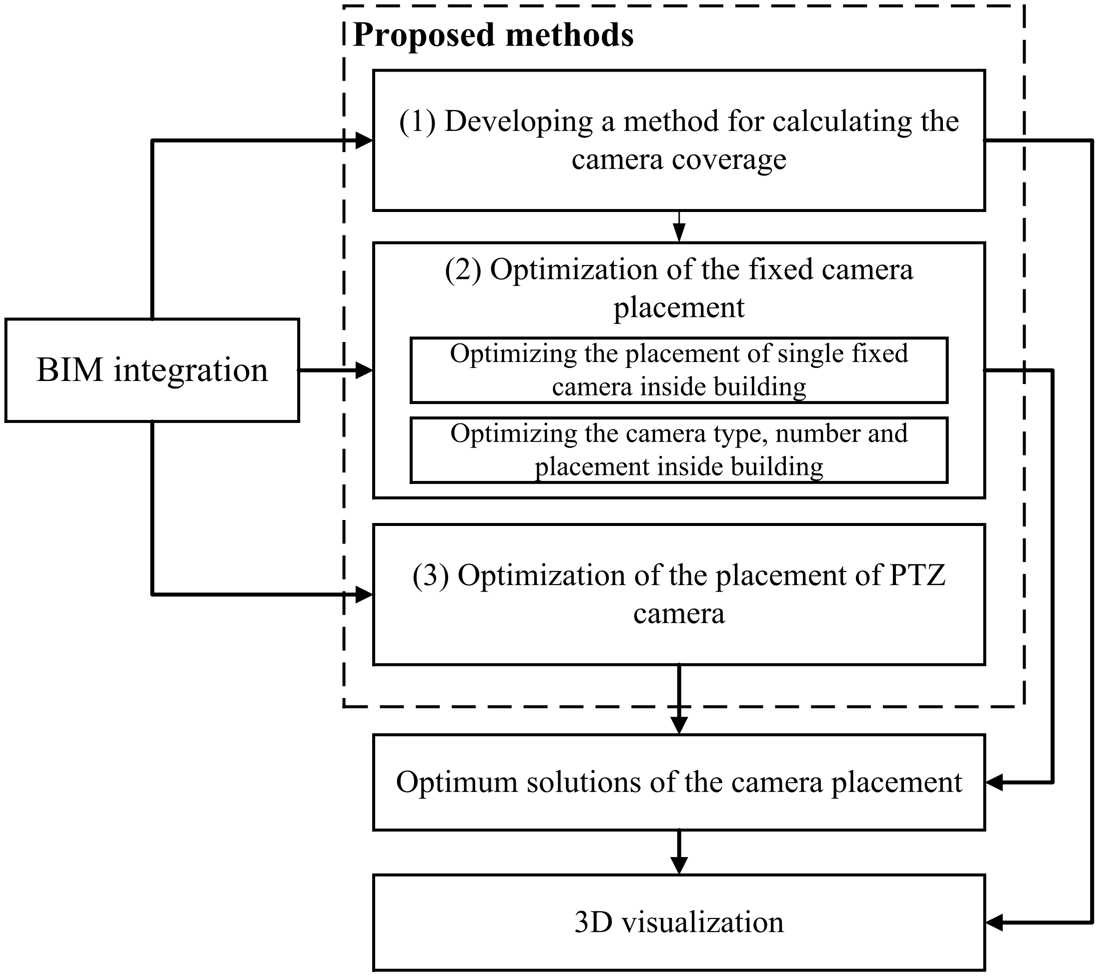


Figure 3-1 Research framework

Method 1: Developing a method for calculating the camera coverage

The objective of this method is to accurately calculate the coverage of fixed cameras inside buildings. The method integrates BIM model with 3D camera models to achieve this purpose. The method aims to automate the coverage calculation process in order to achieve more accurate results than other 2D and 3D techniques. This method will send a BIM model to a game engine tool for the purpose of facilitating the computational process. The method implements a visibility analysis using the concept of ray tracing in order to assist in the process of detecting physical obstacles that can affect the camera coverage. The proposed method will be used in Method 2 and Method 3. This method includes the following:

(1) Developing an integrated BIM model.
(2) Identifying the way of preparing the area that will be covered by the cameras.
(3) Preparing the cameras that will be placed in the building.
(4) Evaluating the visibility of the placed cameras.
(5) Developing the method of calculating the camera coverage.
(6) Validating of the proposed method using a case study.

Method 2: Developing a method for optimizing fixed camera placement in building

(a) Optimizing single fixed camera placement in building

The objective of this method is to optimize the placement of single fixed camera inside building. The method is integrated with the previous method of calculating the camera coverage in order to achieve accurate results that can be used in the optimization process. The proposed method uses GA to solve the optimization problem. The method is using BIM as the main resource that can benefit the optimization process. To declare, BIM can help in specifying the valid camera locations, which will be used in the GA process. This will help to detect obstructing elements that can affect the camera coverage such as geometrical constraints and physical obstacles. More specifically, this method includes the following:

(1) Specifying the search space of the optimization problem using BIM.
(2) Identifying the objective function of maximizing the coverage of a single camera.

(3) Identifying the decision variables in GA.

(4) Generating the candidate solutions.

(5) Integrating the GA optimization with the coverage calculation tool.

(6) Evaluating the candidate solutions in the coverage calculation tool.

(7) Finding the near-optimum solution.

(8) Validating the method using a case study.

(b) Extending the method to optimize the type, number and placement of fixed cameras in building

The objective of this method is to further develop the previous method to optimize the camera type, number and placement. At the end, applying this method can help users to optimize the type, number and placement of cameras required to monitor a specific area inside a building. This method includes the following:

(1) Specifying the search space of the optimization problem using BIM.

(2) Defining the first objective function of maximizing the coverage of multiple cameras.

(3) Defining the second objective function of minimizing the cost.

(4) Identifying the decision variables in GA.

(5) Generating the candidate solutions of multiple cameras' types.

(6) Integrating the GA optimization with the coverage calculation tool.

(7) Evaluating the candidate solutions in the coverage calculation tool.

(8) Evaluating the second objective function of minimizing the cost in GA.

(9) Evaluating the results and finding the optimum Pareto solutions.

(10) Validating the method using a case study.

Method 3: Developing a method to optimize the placement of PTZ camera in building

The aim of this method is to extend the previous proposed method of optimizing the camera placement in order to find the optimum placement of a PTZ camera inside a building. Basically, this will be achieved by finding the optimum cycle path coverage of the camera in a specific area. The method includes the following:

(1) Identifying the cycle path, which ensures acceptable overlapping and logical sequence between phases.
(2) Calculating the coverage-times of phases.
(3) Calculating the cycle path coverage using simulation.
(4) Identifying the objective function of maximizing the cycle path coverage and identifying the decision variables in GA
(5) Generating the candidate solutions of the PTZ camera.
(6) Evaluating the candidate solutions in the simulation tool.
(7) Finding the near-optimum solution.
(8) Validating the method using a case study.

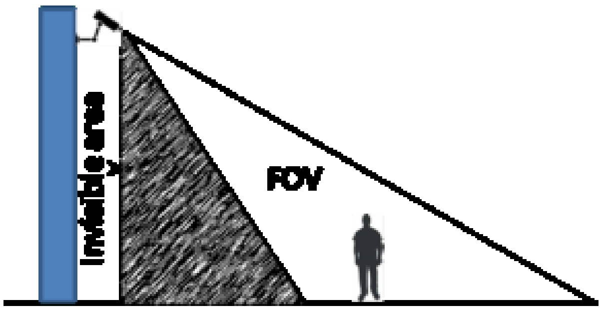
3.3 Summary

This chapter presented the research methodology that aims to develop the following methods of optimizing the camera placement inside buildings: (1) a method for calculating the camera coverage; (2) a method for optimizing fixed cameras' placement inside buildings; and (3) a method for optimizing the placement of a PTZ camera inside buildings.

CHAPTER 4: CALCULATING FIXED CAMERA COVERAGE IN BUILDINGS USING BIM

4.1 Introduction

Although previous studies have implemented different methods to address the fixed camera coverage problem, additional efforts are needed to achieve accurate calculation of the camera coverage. Previous studies implemented 2D or 3D techniques, which have the following problems. First, the usage of the 2D technique may increase the possibility of not covering some important areas which are located under the FOV of the camera, as shown in Figure 4-1 (a), or in places where there are different ceiling or floor elevations. Second, traditional 3D techniques do not consider some building elements that can affect the camera coverage. For example, in Figure 4-1 (b), the installer changed the height of the cameras in order to avoid the shield element and this can affect the camera coverage. The problem is that the installer did not expect to find such an element due to the lack of an efficient method that considers the camera environment before installing the camera. BIM can address most limitations caused by using 2D and 3D techniques in the camera coverage and placement problem. This chapter proposes a novel method that can achieve accurate calculation of the camera coverage inside buildings using BIM. The proposed method will mainly benefit individuals who are responsible for installing cameras inside buildings. This chapter includes three sections. The first section describes the role of BIM and the game engine in the coverage calculation process. The second section details the proposed method including the following: (1) defining the camera placement; (2) validating the camera scenes; (3) generating cells and assigning importance values; (4) visibility analysis; and (5) calculating the camera coverage. The feasibility of the method is demonstrated using a case study. The final section includes the summary and conclusions of this chapter.



(a) Invisible area under the FOV of the camera

(b) Problem of traditional 3D techniques (Chen et al., 2013)

Figure 4-1 The problem of using 2D and 3D techniques

4.2 Role of BIM and game engine in the coverage calculation process

The task of camera coverage calculation can benefit from BIM tool through the following: (1) BIM can consider the location of the light sources that surround the coverage since most parameters of the light sources can be defined in BIM model (Chen et al., 2013); (2) some systems, such as the HVAC components, can generate heat and vibration that might physically affect the cameras. BIM is able to determine the location of these systems and consider them in the camera placement layout; and (3) since it uses a 3D model, BIM can detect invisible areas which are located inside the camera FOV.

This research uses a game engine in the coverage calculation process for the following benefits. The game engine is a platform that can deal with BIM models and apply many physical features (e.g. ray tracing and collision detection), which are already built inside it, to facilitate the implementation of the simulation. Another benefit of using the game engine is the ease of controlling the camera parameters (e.g. FOV and clipping planes) through the adjustment of the camera settings.

4.3 Proposed method for calculating the camera coverage

The study proposes a BIM-based method for calculating the camera coverage inside buildings as shown in Figure 4-2. The method includes five steps as the following: (1) defining the camera placement; (2) validating the camera scenes; (3) generating cells and assigning importance values; (4) visibility analysis; and (5) calculating the camera coverage. Each of the previous steps is detailed in the following sections.

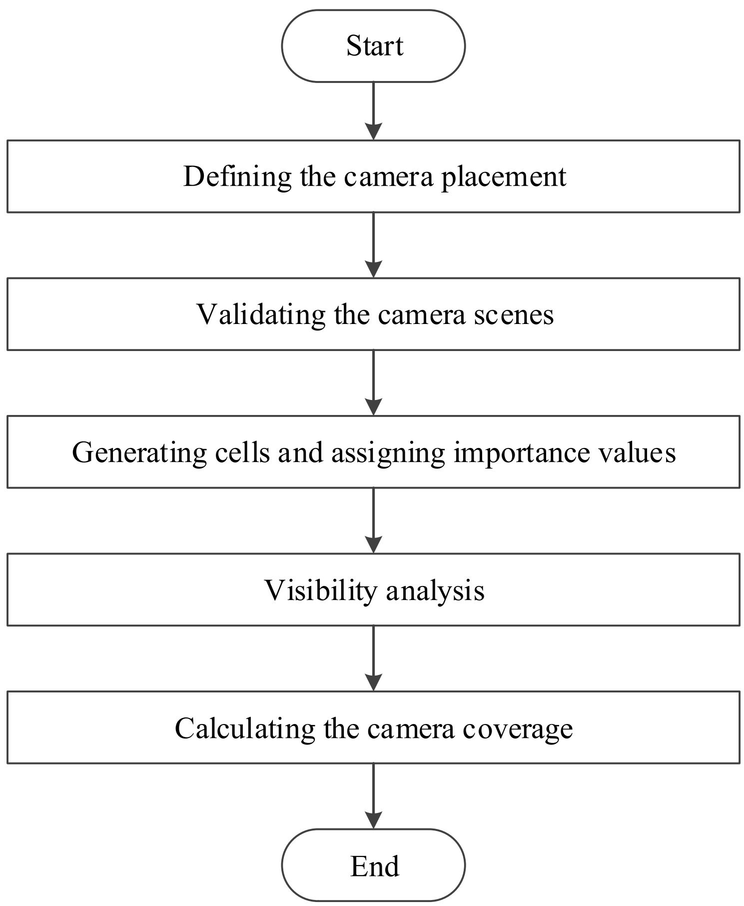


Figure 4-2 Proposed method for calculating the camera coverage

4.3.1 Defining the camera placement

A 2D grid is used to facilitate the process of placing the camera within the boundary of the surface area. Although it is beyond the scope of the current research, it is important to note that changing the size of the 2D grid affects the search space of the optimization process. The smaller grid size will result in a larger search space and longer computational time, but in more accurate results of calculating the coverage, and vice versa. Figure 4-3 shows the method of placing a camera by using the 2D grid.

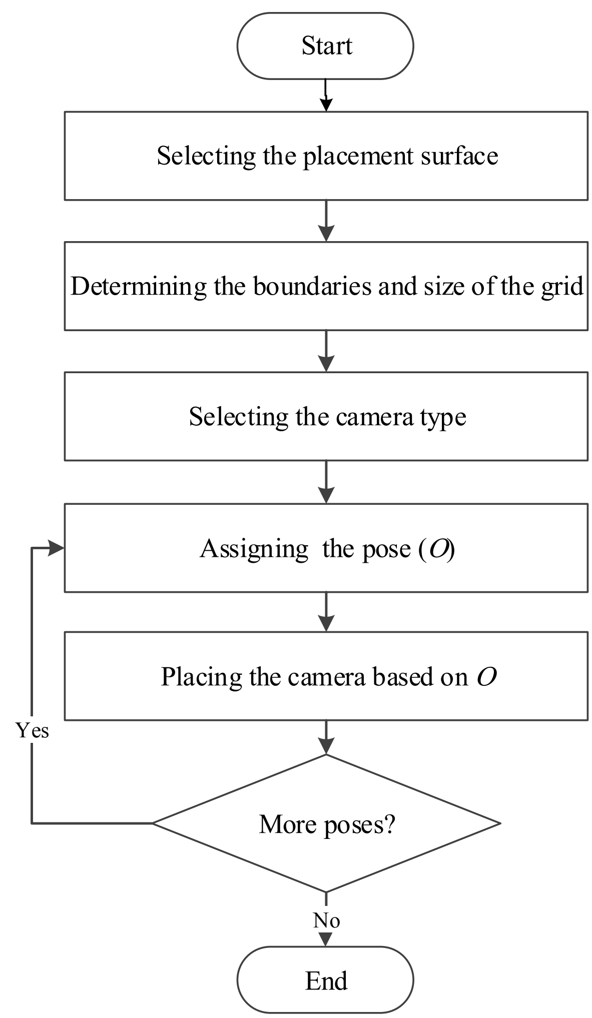


Figure 4-3 Placing camera using 2D grid

It starts by selecting the surface where the camera will be placed and identifying the boundaries and the size of the grid. The user should select the camera type and assign it with a specific pose (i.e. location and orientation). The camera will be then placed based at this pose. This process will be repeated for each camera placement.

Every pose (*O*) comprises the camera coordinates (*X, Y, Z*), which represent the camera location, and the camera *Pan* and *Tilt* angles, which represent the camera orientation. As shown in Figure 4-4, the *Pan* angle (α) represents the camera rotation around a vertical axis with a range between 0° to 360°. While the *Tilt* angle (β) represents the angle with the vertical axis and ranges between 0° and 90°. As a way to decrease the search space, it is necessary to take into account the following rules before placing the camera: (1) all cameras are directed to face the important areas that are inside the BIM model; and (2) the cameras should avoid light sources, shiny and reflective elements.

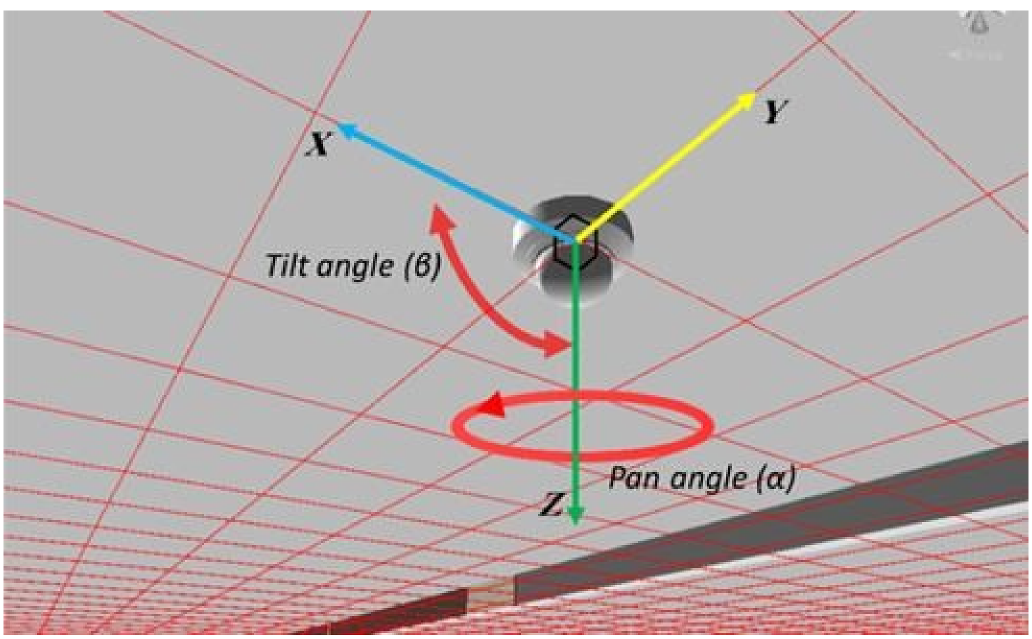


Figure 4-4 Camera orientation and the 2D grid in BIM

4.3.2 Validating the camera scenes

The third step is to examine the validity of the camera pose by evaluating the virtual scene of this camera (i.e. by viewing what the camera can see). This can be achieved by generating 3D virtual scene using the camera FOV and clipping planes.

4.3.3 Generating cells and assigning importance values

This step is focusing on preparing the area that will be used in the calculation of the camera coverage. For this purpose, the floor of the building model is divided into a number of important areas (a_i) that differ in the importance level as shown in Figure 4-5.

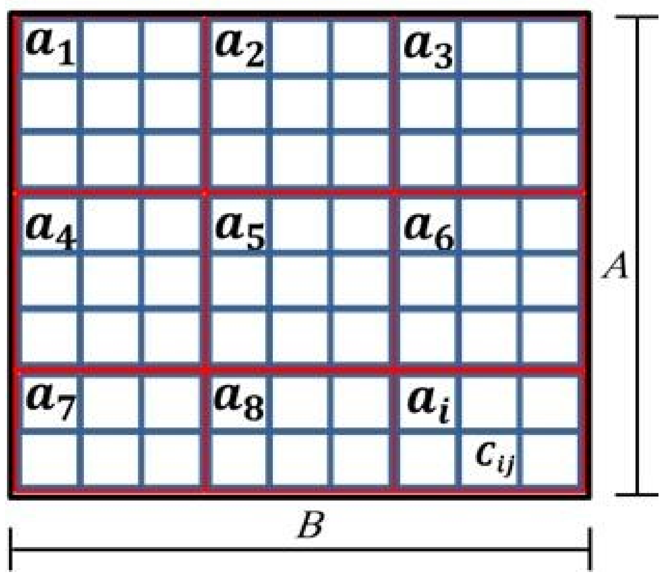


Figure 4-5 Dividing a floor with a number of important areas and cells

As most areas inside buildings have rectangular shapes, cubic cells (C_{ij}) are generated to cover all important areas. It is necessary to find the cell size that can represent the smallest part of the important area. The size of the cell is an important factor that can affect the coverage calculation accuracy. The smaller cell size will increase the number of cells and will increase the computational time but will provide more accurate results, and vice versa. Several layers of cells should be used in the vertical axis of the BIM model to cover the upper limit of the average human height. By covering the monitored areas with cells, it will be possible to determine the volume of important areas that is not covered by the placed cameras.

The numbers of cells along the *X*, *Y* and *Z* axis of the floor (*f*) are calculated by dividing the dimensions of *f* (i.e. *A* and *B*) by the edge length of the cell. In addition, it is important to mention that every generated cell has an importance value assigned by the user based on a set of rules such as the following rules used in the case study: (1) value of 1 is assigned to cells which cover general areas such as corridors; (2) value of 2 is assigned to cells which cover stairs and elevators' halls; and (3) value of 3 is assigned to cells which cover entrances and escalators coming from the underground metro station.

4.3.4 Visibility analysis

The visibility analysis uses ray tracing in order to determine if a ray line is reaching a specific cell or not. Ray tracing can also assist in the process of detecting physical obstacles that might affect the camera coverage. In this study, there are three types of cells: (1) cells forming the structural components of the BIM model such as columns; (2) visible cells; and (3) invisible cells. A cell is considered visible if it is directly hit by the ray line and not hidden behind any element of the BIM model. Ray tracing is used in the visibility analysis as shown in Figure 4-6.

The process starts by generating a ray tracing line R_{x_j} from camera *x* inside its FOV and directed to cell *j*. If R_{x_j} is intersecting with an obstacle, the hidden cells behind this obstacle are highlighted and considered as invisible cells. Otherwise, if R_{x_j} is intersecting with cell *j*, then this cell will be considered with its importance value (*IV*) as a visible cell. It is also important to note that in case the cell is covered by more than one ray line from multiple cameras, the cell will be counted one time in the coverage calculation regardless of the overlapping between the coverages of cameras.

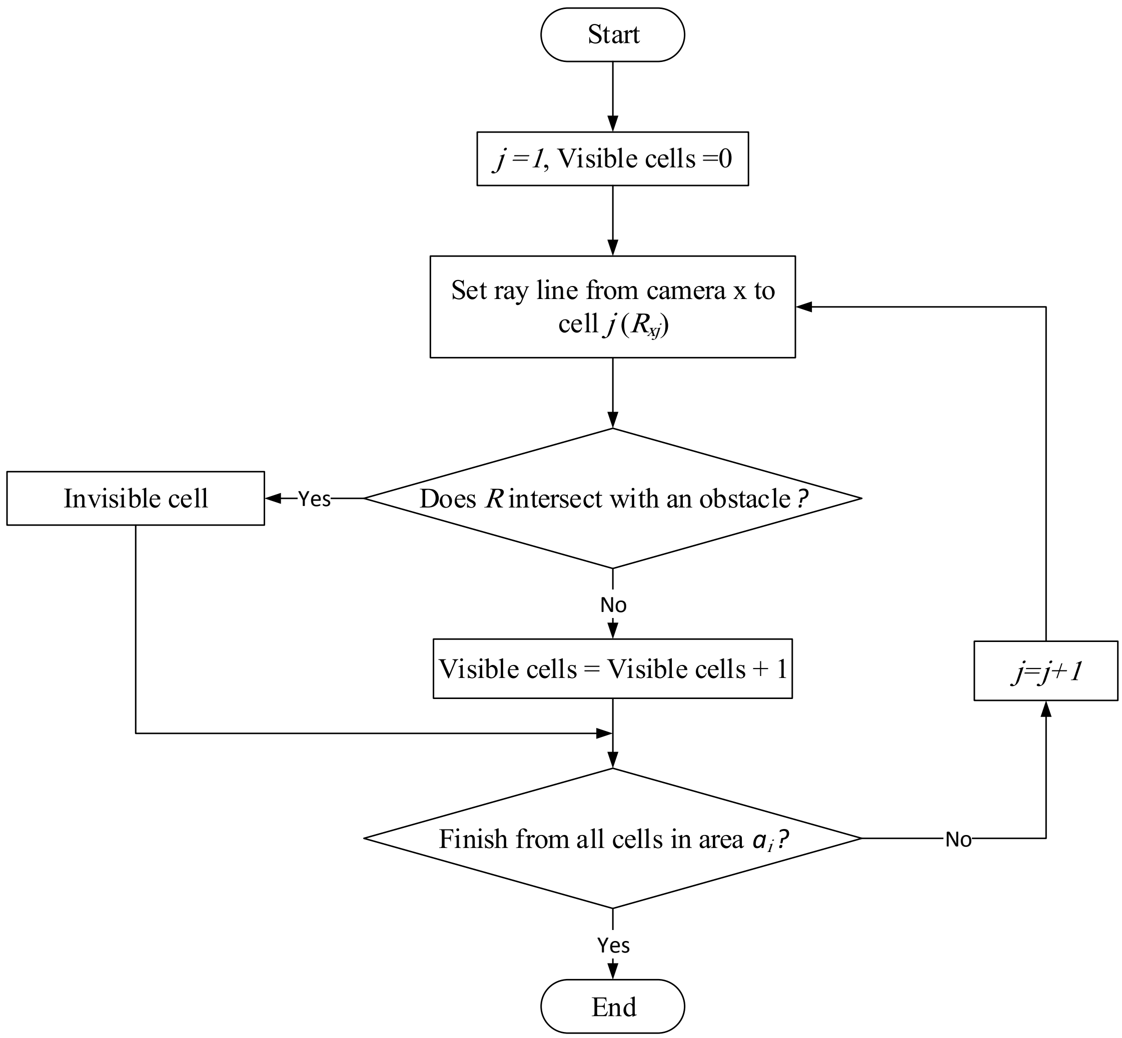


Figure 4-6 Camera visibility analysis in area a_i

4.3.5 Calculating the camera coverage

Figure 4-7 shows a simplified 2D example of camera coverage calculation where a number of important areas (a_i) are divided into cells (C_{ij}) and assigned with importance value (IV_i). Also, a number of cameras (K_x) are placed to cover the important areas. The weighted number of cells in area a_i can be calculated using Equation 4-1. Then, the weighted covered cells (CC_{a_i}) in area a_i is calculated by the summation of all covered cells (C_{iv}) in an important area (a_i) as shown in Equation 4-2. The total coverage of camera (K_x) is calculated using Equation 4-3 (Amriki and Atrey, 2012). This is done by the summation of the weighted covered cells in all areas divided by the summation of all weighted cells in the monitoring area. Finally, the total coverage of all placed cameras can be calculated using Equation 4-4.

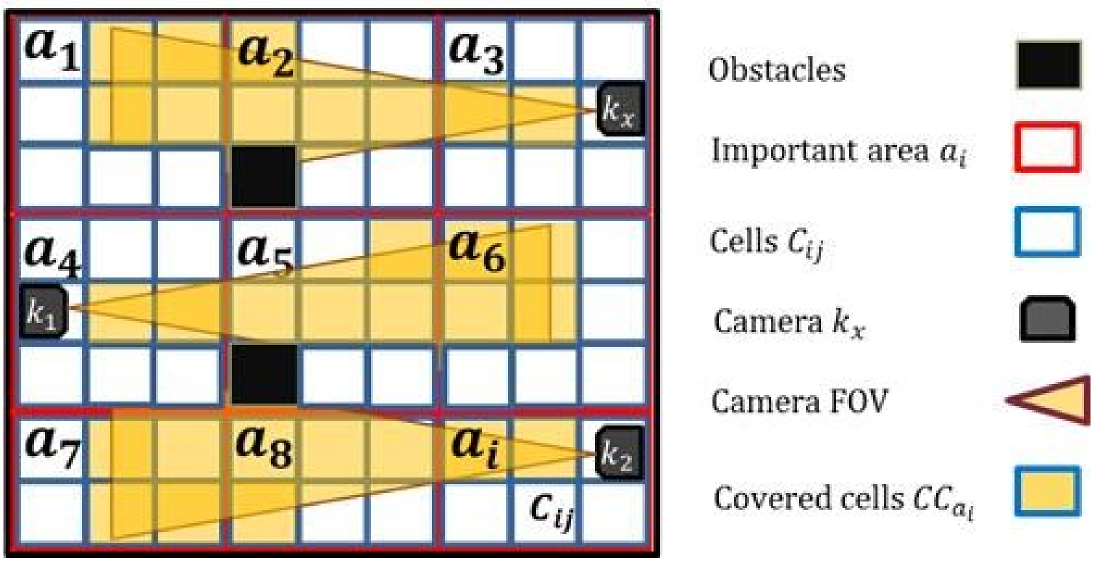


Figure 4-7 Example of coverage calculation

$$W_{a_i} = IV_i \sum_{j=1}^{n} C_{ij} \qquad \text{Equation 4-1}$$

$$CC_{a_i} = IV_i \sum_{v=1}^{n`} C_{iv} \qquad \text{Equation 4-2}$$

$$Total\ coverage\ of\ camera\ K_x = \frac{\sum_{i=1}^{m} CC_{a_i}}{\sum_{i=1}^{m} W_{a_i}} \qquad \text{Equation 4-3}$$

$$Total\ coverage\ of\ all\ cameras = \sum_{x=1}^{Q} Coverage\ of\ camera\ K_x \qquad \text{Equation 4-4}$$

where W_{a_i} is the weight of important area a_i, C_{ij} represents the cell j in important area i, and j = 1:n, IV_i is the importance value assigned to cell j in area i, and i = 1:m, C_{iv} is the covered cell v in area a_i and v = 1:n`, K_x is camera x and x = *1: Q* and CC_{a_i} is the weighted covered cells in area a_i.

4.4 Implementation and case study

A prototype system has been developed to demonstrate the feasibility of the proposed method. The case study is implemented on the ground floor of the Engineering and Visual Art (EV) building in Concordia University. This case study is demonstrating the applicability of the proposed method using BIM technology for evaluating the coverage of a number of cameras placed in this floor. Seven fixed cameras of various types (single and multiple camera types) cover specific locations that include important and unimportant areas. In addition, the floor has some physical constraints such as columns and stairs that can affect the camera coverage. The BIM model was developed using Revit and integrated with seven 3D camera models (Autodesk Revit, 2014) for the purpose of simulating the camera placement layout. These camera models are provided by several manufacturers, as shown in Figure 4-8. Next, is to import the integrated BIM model into the game engine. This step includes two processes. The first process is integrating the BIM model with 3D models of surveillance cameras available from the camera manufacturers. Each camera has different parameters that affect the FOV and the distance between the clipping planes. Clipping planes define the space in which an object can be seen and rendered based on the distance between these planes. Parameters that affect the coverage are the focal length, the height of the camera, the height of the target and the distance between the camera and the target (Chen et al., 2013). The integrated BIM model is imported to the game engine to facilitate the computational process. The second process is to align the FOV of a virtual camera available in the game engine with each FOV of the imported cameras in BIM. The purpose of this alignment is to simulate the parameters of the BIM FOVs in the game engine. The virtual FOVs of the game engine are used to validate the virtual scenes (Section 4.3.2) and to serve the visibility analysis (Section 4.3.4).

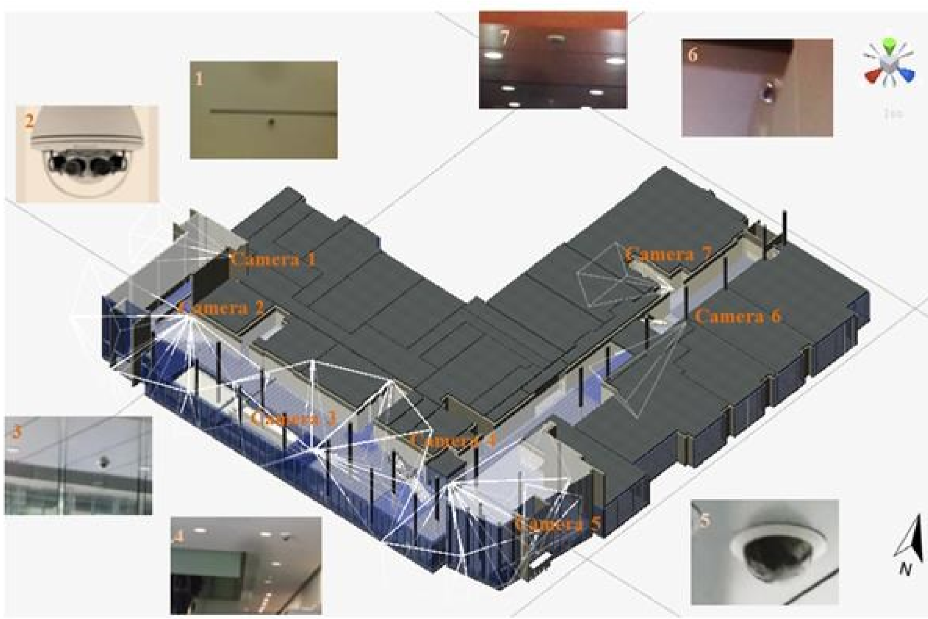


Figure 4-8 BIM model aligned with FOVs in the game engine

The focus in this case study is on Camera 3. Multiple camera placements will be considered in our future work. Then, the integrated BIM model is imported into the game engine of Unity3D (Unity, 2015) using FBX format in order to facilitate the computational processes which were developed using C# language. Inside Unity3D, the process of generating cells over the studied area was developed in order to evaluate the camera visibility. These cells can help to obtain accurate results of the camera coverage by changing their size as will be explained later. Also, the process of generating a 2D grid system is developed to facilitate the placement of additional surveillance cameras on the ceiling if needed. Seven virtual FOVs were generated in Unity3D and aligned with the cameras of the BIM model to simulate the current status of camera placement in this floor; using the information of the BIM camera models taking into account their parameters. The parameters of the virtual FOVs are adjusted to precisely match the cameras of the BIM model as shown in Figure 4-8. These parameters are FOV angle, camera height, and nearest and farthest point. The virtual FOVs will help to generate the virtual scenes and to apply the visibility analysis using the ray tracing concept as shown in Figures 4-11 and 4-13, respectively.

The next step is to generate a 2D grid inside BIM model. This grid is generated in order to facilitate the placement of the cameras. The shorter distance between the grid lines will offer more accurate results and vice versa. In this case study, a number of lines is generated in each direction (i.e. *X* and *Y* directions) on the ceiling surface. In the case study, the user determines 100 lines in the *X* and *Y* directions of the grid. The process of generating the 2D grid system starts by selecting the ceiling that will be covered with the grid system. Next, a box collider available in Unity3D is added to the ceiling where the camera will be placed. The collider surrounds the ceiling shape for the purpose of determining the borders of the grid. Finally, the user can select and place additional 3D cameras on the grid. Another step is to identify the areas that will be targeted by the selected camera. In this case study, 17 important areas were specified. Then, a number of cells were generated over these important areas as shown in Figure 4-9. The process of generating cells was implemented using the following steps: (1) selecting the floor shape that will be covered with cells; (2) creating a mesh collider that determines the shape of the floor; (3) determining the suitable size of the cells; (4) determining the number of cells on the *X, Y, Z* axis of the floor shape; (5) specifying the importance value of each cell based on its important area a_i; and (6) running the cell generating process.

Figure 4-9 Plan view of the BIM model showing the important areas with their Importance Values (*IV*)

As previously discussed, cells are generated in a way to cover the upper limit of the average human height. The code of generating cells in the game engine is given in

Appendix B. A sensitivity analysis is conducted to evaluate the suitable edge size of the generated cells to cover two meters height, as shown in Table 4-1. The evaluation starts by generating cells with a side dimension of 0.1 m to cover an area of 10 m by 7 m, as shown in Figures 4-10 (a) and (b). The evaluation showed that the required number of cells to cover this area was 163,500 with 20 layers of cells. After conducting the camera coverage calculation process, it showed that the camera required 4.14 seconds to cover 81.39 m^3 of cells volume. The value of covered cells was calculated using Equation 4-5. Also, the change of the accuracy was calculated to evaluate different sizes of cells from 0.1m to 1m by using Equation 4-6.

$$CCV = Number\ of\ covered\ cells \times Cell\ volume \qquad \text{Equation 4-5}$$

$$C_{ac} = \frac{Current\ CCV - First\ CCV}{First\ CCV} \qquad \text{Equation 4-6}$$

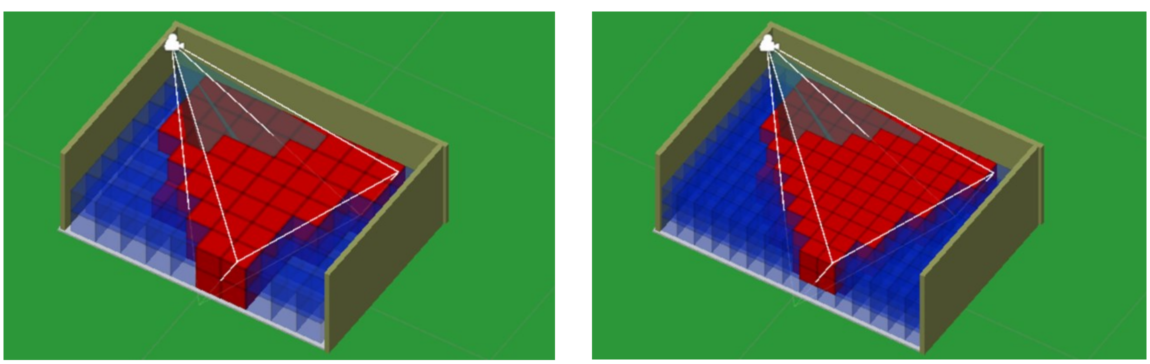
Where CCV is the covered cell volume and C_{ac} is the change in accuracy

Cells with an edge of 0.7 m were selected as the suitable cell size because the result of covered cells volume was 79.91 m^3 with the least change of accuracy of 1.82% compared with the 0.1 m cell size. This cell size also showed that the time of the cell generation process and the time of camera coverage calculation were 0.09 second and 0.1 second, respectively. After generating the cells, it is necessary to classify the important areas based on the importance values of their cells. Table 4-2 shows the number of generated cells over each important area assigned with their importance values.

Camera 3 was selected to evaluate the coverage and to generate 3D virtual scenes as shown in Figure 4-11. It is important to note that Camera 3 is a multiple camera type which includes 4 lenses and that generate 4 FOVs.

Table 4-1 Selecting the suitable cell size

Cell size (m)	*# of layers*	*# of cells*	*Covered cells volume (m^3)*	*Change in accuracy (%)*	*Time to generate cells (sec.)*	*Coverage calculation time (sec.)*
0.1	20	163,500	81.39	0	2.25	4.14
0.2	10	19,980	79.67	-2.12	0.16	0.38
0.3	7	6,300	83.80	2.96	0.05	0.22
0.4	5	2,430	79.55	-2.26	0.02	0.10
0.5	4	1,260	79.80	-1.96	0.01	0.05
0.6	4	864	93.96	15.43	0.01	0.02
0.7	3	450	79.91	-1.82	0.09	0.01
0.8	3	351	92.16	13.22	0.08	0.01
0.9	3	288	102.06	25.38	0.08	0.01
1.0	2	140	73.00	-10.31	0.07	0.01



(a) Generating 2 levels of cells (b) Generating 3 levels of cells

Figure 4-10 Sensitivity analysis of the suitable cell size

These virtual scenes can help to validate if Camera 3 is properly placed by viewing what the camera can see in an animation mode. These virtual scenes can be automatically generated by selecting the FOVs of Camera 3 and specifying the screen dimensions of each FOV. This evaluation will show if there are any objects that may affect the visibility of this camera.

Table 4-2 Number of generated cells in each area

Area	*Importance value*	*Description*	*Number of cells*
a_1	3	Area surrounding entrance 1	405
a_2	1	General area facing entrance 1	311
a_3	2	Area connects this building with another building	307
a_4	1	Area of corridor close to entrance 1	466
a_5	1	Area of corridor leads to elevators	378
a_6	2	Area surrounding elevators	231
a_7	3	Area facing escalators	105
a_8	1	General area facing elevators and escalators	663
a_9	2	Area surrounding stairs	428
a_{10}	1	General area facing entrance 2	1,782
a_{11}	1	Area surrounding entrance 2	119
a_{12}	3	General area leads to entrance 3	688
a_{13}	1	Area surrounding stairs and leads to entrance 3	312
a_{14}	2	Area surrounding stairs and leads to entrance 3	331
a_{15}	2	Area surrounding elevators	217
a_{16}	2	Area close to entrance 3	365
a_{17}	3	Area close to entrance 3 (different floor elevation)	204
		Total number of cells	7,312

An experiment was conducted in order to examine the impact of FOV angle and the tilt angle (β) on the visibility of Camera 3. The first step is to determine the areas that will be covered by the camera. In this experiment, Camera 3 is targeting area 5 to area 8. The four areas are covered by 1,377 cells, as shown in Figure 4-12. These cells are assigned with Importance Values (*IV*) as the following: (1) cells covering areas a_5 and a_8 that represent general areas are assigned with *IV*=1; (2) cells of area a_6 that represents elevator hall are assigned with *IV*=2; and (3) cells covering area a_7 that represents the escalator location connecting to the metro station are assigned with *IV*=3. Camera coverage is calculated using Algorithm 4-1 based on the following aspects: (1) The visibility process is conducted through all important areas within the camera FOV; (2) The weight of the total monitoring area is calculated by multiplying the total number of generated cells by the importance values assigned to every important area; (3) Visible cells are considered with their importance values in order to be used in calculating the coverage for every important area; and (4) The total camera coverage is calculated by the summation of the coverage for every important area inside the camera FOV. The code of calculating the coverage is attached in Appendix D.

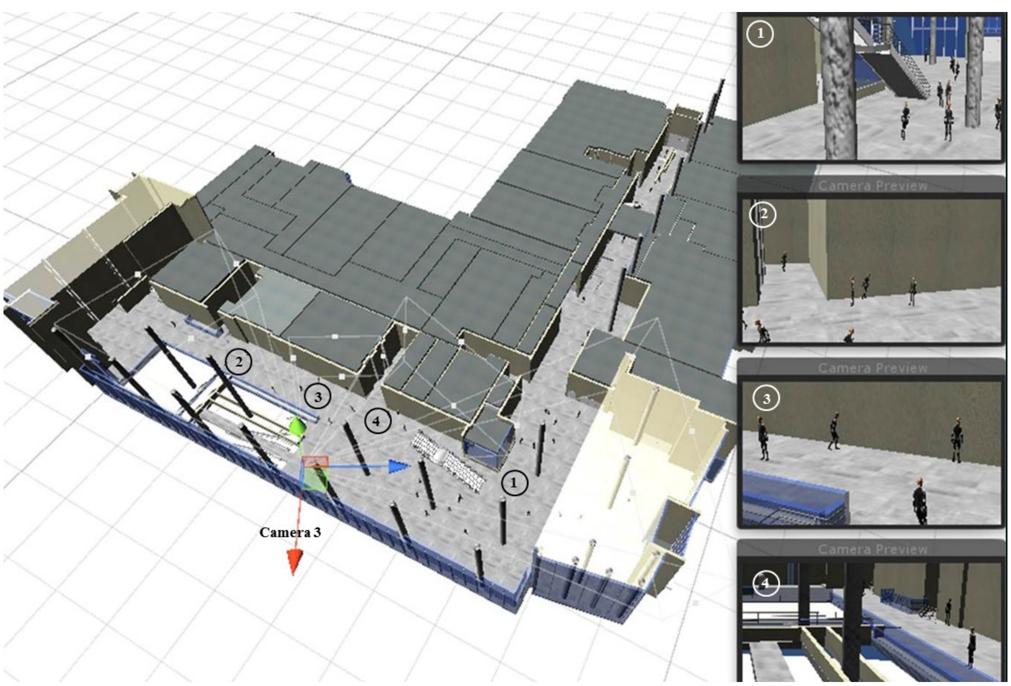


Figure 4-11 Generating virtual screen views

Algorithm 4-1: Calculating the camera coverage

```
total_camera_coverage = 0 ;
for i = 1:m do
CCa[i]= 0 ;
Wa[i]= 0 ;
  if area[i] intersectWith FoV
    for j = 1:n do
      if cell[j] intersectWith FoV
        for v = 1:n' do
          if cell[v] isVisible
            CCa[i]= CCa[i]+ C[i][v] x IV[i] ;
            cell_material_colour = red ;
          else
            cell_material_colour = green ;
          end if
        end for
      else
        cell_material_colour = no_color
      end if
      Wa[i]= Wa[i]+ C[i][j] x IV[i] ;
    end for
    camera_coverage[i] = CCa[i] / Wa[i] ;
  else
    camera_coverage[i] = 0;
  end if
  total_camera_coverage = total_camera_coverage + camera_coverage[i] ;
end for
```

Figures 4-13 (a) to (d) show the four scenarios for evaluating the coverage of Camera 3 as the following: (a) FOV = 35° and β = 0°; (b) FOV = 35° and β = 10°; (c) FOV = 40° and β = 0°; and (d) FOV = 40° and β = 10°. The red color represents the visible cells while the green color represents the invisible cells behind a column. Table 4-3 shows the coverage results of the four scenarios. Implementing scenario *a* gave the coverage of 63.6%. By adjusting β to 10° in scenario *b*, the coverage became 87.12%. The coverage decreased when increasing the FOV to 40° while fixing β in scenario *c*. However, in scenario *d* when adjusting β to 10° and the FOV to 40°, the coverage of 89.4% was achieved. This analysis demonstrated that changing the FOV and β can impact the camera coverage results. The results showed the importance of using the 3D method and considering the camera height during the process of calculating the coverage in order to detect invisible spots which was simulated by the green cells.

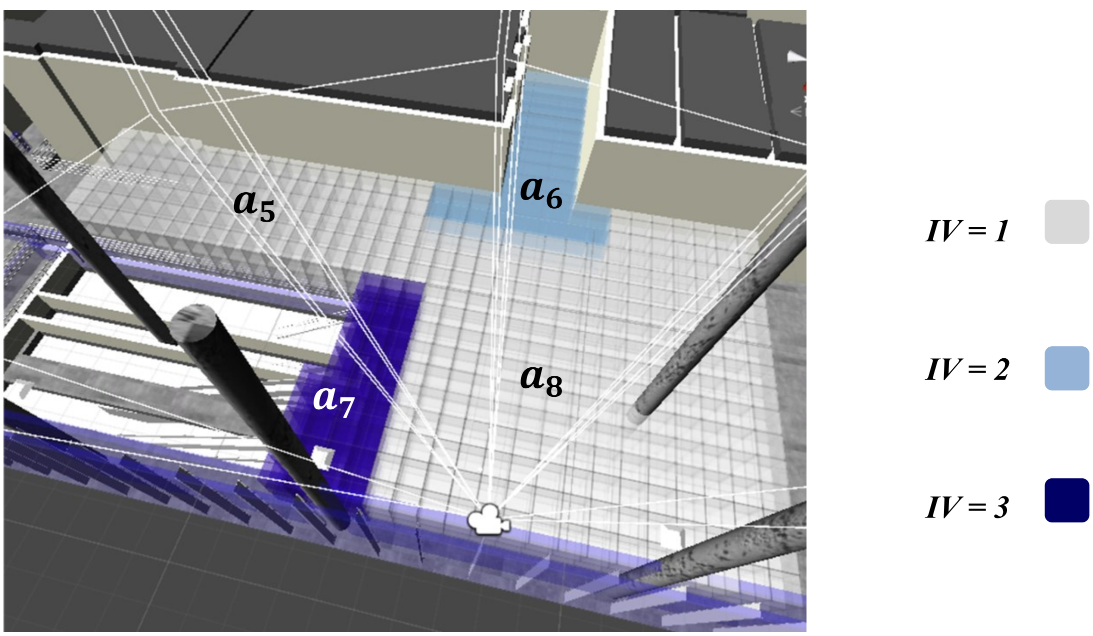
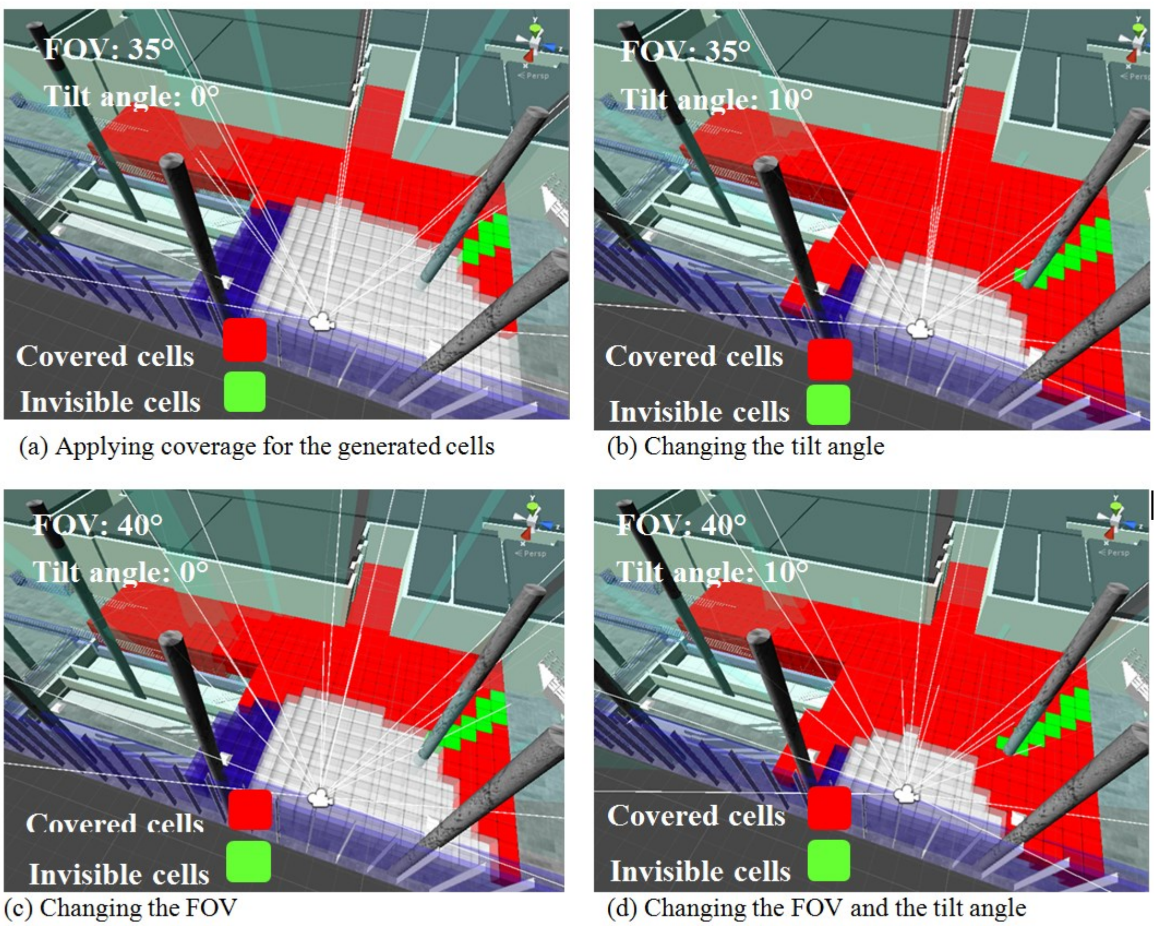


Figure 4-12 Cells assigned with importance values in areas a_5 to a_8

Table 4-3 Scenarios of the camera coverage for areas a_5-a_8

# Scenario	*β(°)*	*FOV (°)*	CV_5	CV_6	CV_7	CV_8	COV_5 *(%)*	COV_6 *(%)*	COV_7 *(%)*	COV_8 *(%)*	*Total coverage of Camera 3 (%)*
a	0	35	378	231	12	282	100	100	11.4	42.5	63.60
b	10	35	378	231	87	483	100	100	82.8	72.8	87.12
c	0	40	378	231	24	303	100	100	22.8	45.7	66.80
d	10	40	378	231	93	507	100	100	88.5	76.4	89.40



(a) Applying coverage for the generated cells (b) Changing the tilt angle

(c) Changing the FOV (d) Changing the FOV and the tilt angle

Figure 4-13 Implementing scenarios on Camera 3

4.5 Summary and conclusions

This chapter presented a new method for calculating the camera coverage inside buildings using BIM model in order to automate the calculation process and to achieve accurate results. The proposed method included: (1) defining the camera placement, (2) validating the camera scenes, (3) generating cells and assigning importance values, (4) visibility analysis, and (5) calculating the camera coverage. The chapter also included a sensitivity analysis of evaluating the suitable cell size which is used to cover the monitored area. The case study demonstrated the benefits of using BIM model in calculating the camera coverage in buildings. Four scenarios conducted to evaluate the effect of the FOV and β angles on the camera coverage. The results showed the ability of the FOV and β angle to increase the coverage to 89.40% in scenario (d). The experiment proves the necessity of considering the height of the camera during the process of calculating the coverage in order to detect invisible spots behind obstacles. Also, dividing the monitored area into a number of important areas with different (*IV*) values facilitated the decision of selecting the proper FOV and β angles for the camera. The main contribution of this chapter is a method that can calculate the camera coverage inside buildings taking into account the existence of different obstructing elements that can affect the camera coverage.The proposed method is beneficial for installers to find accurate calculation of the camera coverage before deciding its suitable placement.

CHAPTER 5: SIMULATION-BASED OPTIMIZATION OF FIXED CAMERAS TYPES, NUMBER AND PLACEMENT IN BUILDINGS USING BIM

5.1 Introduction

The objective of this chapter is to find the optimum cameras' type, number and placement inside a building in a way to ensure the maximum camera coverage with the minimum cost. A simulation-based optimization method is developed using BIM and GA to solve the optimization problem. The objective function of this algorithm can be a single objective function (i.e. camera coverage), or multi-objective functions (i.e. camera coverage and cost). The camera is placed on the ceiling or the wall surface and restricted by three major factors: (1) grid system that matches the ceiling size or the upper part of the wall; (2) boundaries that can limit the coverage (i.e. walls); and (3) geometrical constraints located on the placement surface. On the other hand, evaluating the camera placement requires considering the following elements: (1) the relationship between the placed cameras and their surroundings, specifically the important areas that should be covered; and (2) the type of the used cameras including their features, capabilities and limitations.

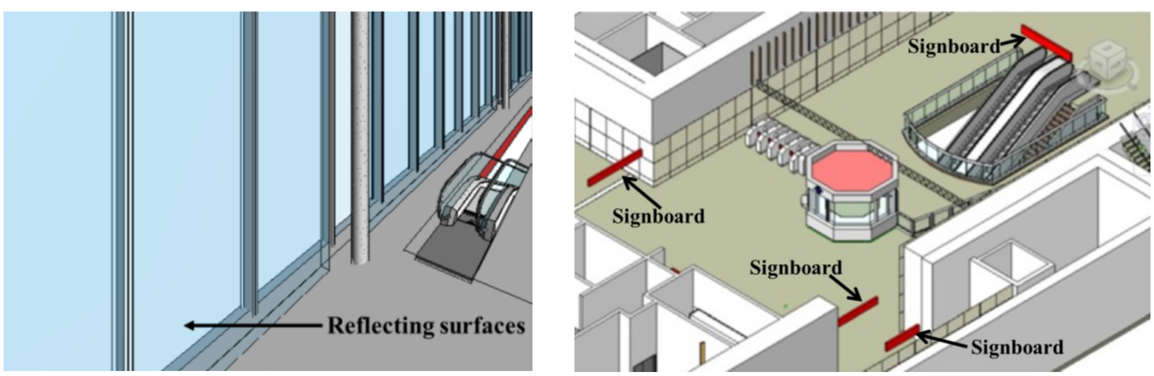
The method is integrated with the previous proposed method of calculating the camera coverage in order to achieve accurate results that can be used in the optimization process. More specifically, the previous method can serve this chapter in the following: (1) receiving the candidate solutions created by the optimization process; (2) allocating the camera based on the given solutions; and (3) calculating the cameras coverage and returning the results to the optimization process.

This chapter includes the following: (1) detailed description of the simulation-based optimization method of the camera placement; (2) implementation and case study to evaluate the proposed method; and (3) the summary and conclusions of this chapter.

5.2 Simulation-based optimization method of the camera placement

5.2.1 Role of BIM in the optimization process

During the optimization process, BIM can be used to define the input of the optimization due to the following two reasons. First, building information about geometrical constraints, physical obstacles and different ceiling and floor elevations are basic factors that can affect the optimization process. For example, reflecting surfaces or ceiling signboards, as shown in Figures 5-1 (a) and (b), can affect the camera coverage. BIM tool can play a vital role in providing detailed information about these factors in order to consider them during the camera placement.



(a) Reflecting surfaces (b) Ceiling signboards (Chen et al., 2013)

Figure 5-1 Factors affecting the camera coverage

Second, BIM can be used in specifying the search space of the optimization process by identifying the valid locations of the camera placement over the placement surfaces (e.g. ceilings and walls). It is important to mention that because of the ability of BIM tool to support the concept of interoperability, the resulting 3D BIM models at the end of the optimization process can also help different stakeholders to evaluate the design of the camera placement in case they need to conduct adjustments on the building details. This would help in reducing the impact of these adjustments on the camera coverage. Unlike other tools such as Geographical Information Systems (GIS), which are limited to spatial information of outdoor spaces, BIM can be used to automatically provide geometrical and non-geometrical attributes of these constraints in order to be considered in the optimization process.

As shown in Figure 5-2, BIM is applied in the optimization process in the following steps: (1) BIM tool is used to specify the search space (i.e. valid locations of the cameras on the placement surface). This is done by identifying the types of obstructing elements and excluding their locations; (2) The valid locations are sent to the optimization process to start generating the solutions (i.e. poses) of the camera placement; (3) These solutions are sent to a 3D simulation process, which also uses the BIM model, in order to calculate the coverage. This process includes preparing the cameras and the monitored areas in order to conduct the visibility and the coverage calculation process; (4) The coverage results are then returned to the optimization process in order to be evaluated; and (5) The final optimum solutions are visualized using the 3D simulation process.

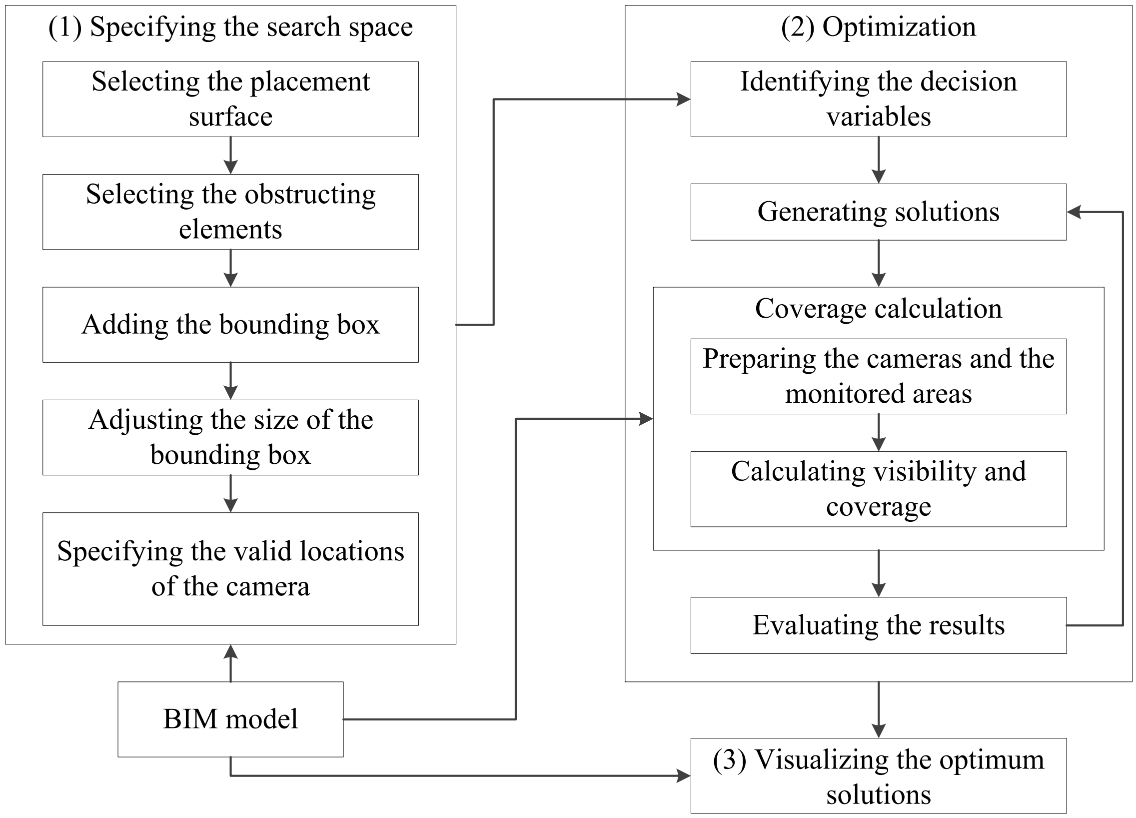


Figure 5-2 Role of BIM in the optimization process

5.2.1.1 Factors Affecting the Optimization Process

The optimization process can be affected by the following factors: (1) The surface where the camera is placed (e.g. ceiling) considering the height to the placement surface. (2) The important areas to be covered considering the height of the monitored area (i.e. the upper limit of the average human height). The study classifies the important areas as the following: (a) all entrances and exits of the building since all users of the building pass through these areas; (b) the common areas, which include high people density such as corridors, especially areas where there is an intersection between more than one corridor; and (c) elevators, stairs and escalators locations since they include people movement from one floor to another. (3) The existence of private areas such as bathrooms. (4) The existence of constraints located on the surface of the camera placement.

It is necessary to understand the impact of these constraints on the camera coverage and the role of BIM in considering them during the optimization process. For example, the HVAC components can affect the camera placement due to their ongoing vibration. This vibration can affect the video quality and the internal components of the camera, which will affect its life span. It is also necessary to determine the locations of these constraints and to consider them during the camera placement process.

5.2.1.2 Defining the Constraints and Search Space Using BIM

Using BIM in specifying the constraints will help to identify the valid locations (i.e. points) on a grid attached to the camera placement surface. The points of the grid are used to define the search space of the optimization process. It is possible to reduce the time of the optimization by limiting the camera placement to the valid locations and avoiding invalid locations that are affected by the constraints. There are four types of constraints: (1) geometrical constraints attached to the ceiling (e.g. ceiling signboards); (2) operational constraints close to the camera location (e.g. the vibration produced by HVAC components); (3) logical constraints (e.g. the camera faces a wall or reflecting surfaces); and (4) legal constraints (e.g. the camera is viewing private areas).

In BIM, the geometrical and non-geometrical attributes (e.g. type, length and location) can be used to automatically specify these constraints on the surface where the camera will be placed. Figure 5-3 shows an example of specifying the search space. BIM is used to create buffers that include the impact areas of these constraints (e.g. vibration or heat sources). This is done by assigning a bounding box (i.e. buffer) with a specific size, which covers the impact area. It is not easy to have a definite number for the size of the buffers. Three experts in surveillance cameras outlined that the size of the buffers is decided based on experience by defining several levels of the constraint effect. For example, the effect of vibration can be specified as low, average and high. Further work will aim to identify the buffer size using a survey. The points outside these buffers are considered as the valid locations of the camera placement.

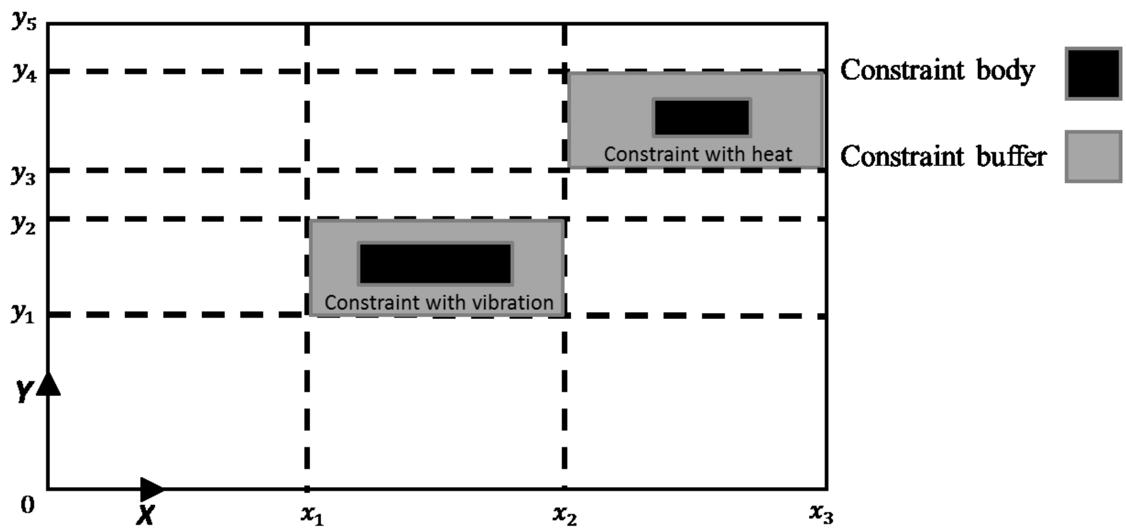


Figure 5-3 Example of specifying the search space using BIM model

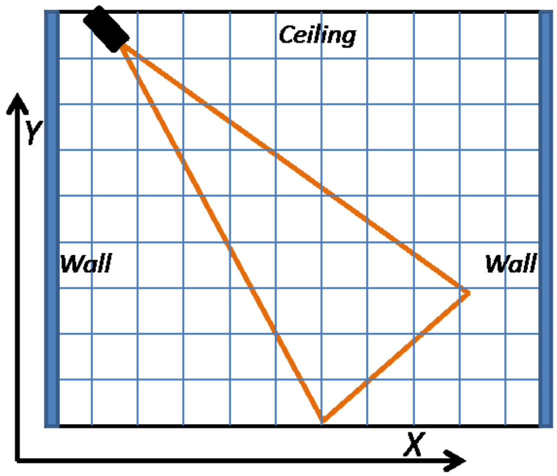
5.2.2 GA optimization method for the camera placement

Finding the optimal configuration of multiple sensors to cover the maximum total area is a combinatorial optimization, which is considered as NP-hard problem (Cole and Sharir 1989). This is because the number of solutions exponentially increases when the camera configurations are increasing (Murray et al. 2007). In order to address the optimization problem, this research adopted GA optimization, which has the following benefits over other techniques: (1) GA can be used to identify the global near-optimum tradeoffs and do not tend to be stuck at a local optimum (Bandyopadhyay and Pal, 2007); (2) GA is insensitive to the shape of the Pareto front (Goh et al., 2009); and (3) GA is easy to implement, and could be conducted in a parallel environment (Abraham et al., 2005).

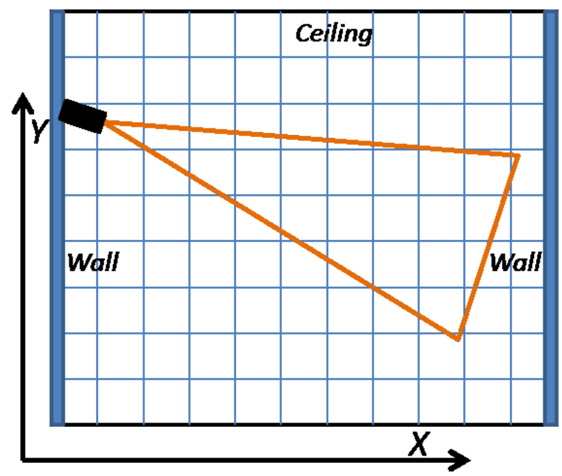
GA has been previously used for the camera placement problem. For example, Kim et al. (2008) implemented a multi-objective genetic algorithm for surveillance sensor placement. The outcomes showed the applicability of the multi-objective GA to find a set of non-dominated solutions of the sensor placement. This research is using the Preference-Inspired Co-evolutionary Algorithm (PICEA) (Purshouse et al. 2011) for the purpose of obtaining Pareto front solutions that capture the trade-off between the coverage and cost of different cameras. The algorithm generates a population of solutions for a specific number of generations. This section includes the following: (1) Identifying the decision variables, (2) The role of GA in the camera placement, and (3) Simulation-based optimization process.

5.2.2.1 Identifying the Decision Variables

The first step of the optimization process is to identify the location (i.e. point on the grid system) and orientation of the camera by determining the location variables (*X, Y* and *Z* coordinates) and the orientation variables (*Pan* and *Tilt* angles). These variables help in placing the camera on the surface. As explained in Chapter 4, the 2D grid is used to facilitate the placement of the cameras within the boundaries of the placement area. Although it is clear that there are five variables for each camera pose (*O*), some of these variables will be fixed based on the case of camera placement. For example, in case of placing the camera on the ceiling of a room, as shown in Figure 5-4 (a), the *Y* coordinate will be fixed to match the height of the room. On the other hand, the *X* coordinate is fixed in the case where the camera is placed on the wall, as shown in Figure 5-4 (b).



(a) Section view of camera placement on the ceiling with fixed Y value



(b) Section view of camera placement on the wall with fixed X value

Figure 5-4 Examples of camera placement

5.2.2.2 The Role of GA in the Camera Placement

In order to understand the role of GA parameters in the camera placement process, the following example is given as shown in Figure 5-5. In this example, the number of generations (G) is assumed to be 100. One generation (g) has a population size (P) of 9 genes (individuals). Each gene is a solution of the GA that gives different numbers, types and poses of cameras inside a specific space. Each gene includes 1 to 6 cameras. Each camera pose (O) has five variables X, Y, Z, *Pan* and *Tilt* angles. In case of multiple cameras placement, each camera will have the same variables in addition to the FOV value, which represents the type of the camera among a fixed number of available types.

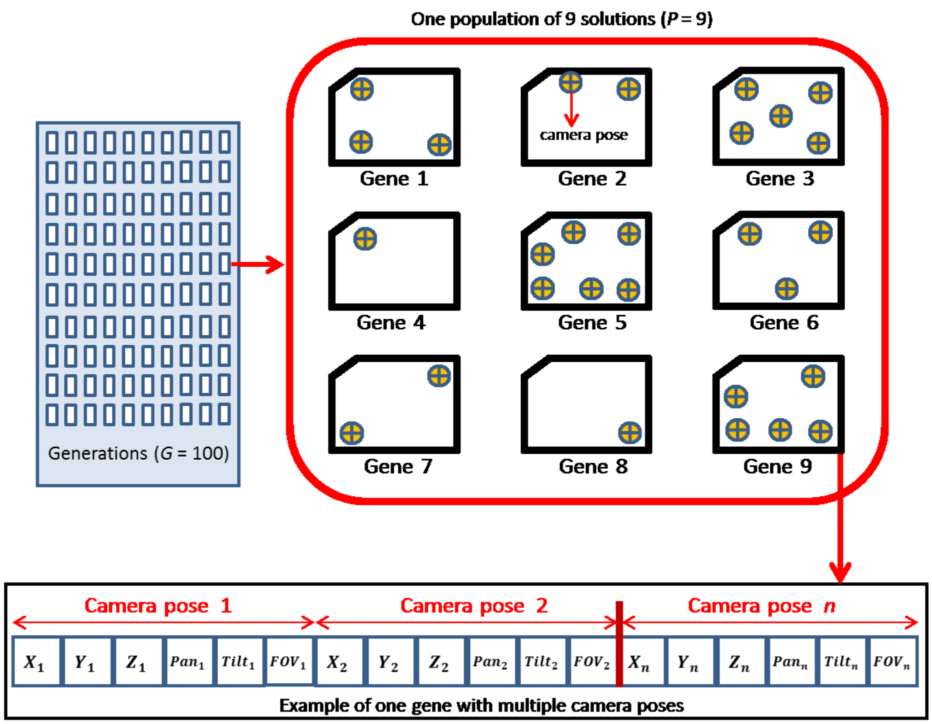


Figure 5-5 Example of the GA genes representing the camera poses

Furthermore, it is important to understand the effect of the population size on the camera placement optimization. The bigger the population size, the more accurate the results and the longer the computation time, and vice versa. Other parameters, including the crossover and mutation rates, should be also determined in the GA. If the number of placed cameras is more than one (e.g. three cameras) and the number of variables of each camera is 6 (i.e. *X, Y, Z, Pan* and *Tilt* angles and FOV), then in case of placing three cameras on the ceiling, the total number of variables will be 15 in addition to three fixed *Y* coordinates on the ceiling.

5.2.2.3 Simulation-Based Optimization Process

As previously discussed, BIM is used to specify the search space by identifying the valid locations on the placement surface and excluding the areas that are affected by the constraints. Then, these valid locations are sent to the optimization process to generate the random solutions. As shown in Figure 5-6, the GA process starts by initializing the first generation (*g*). This will lead to initialize the first population, which includes random solutions of the camera placement (genes).

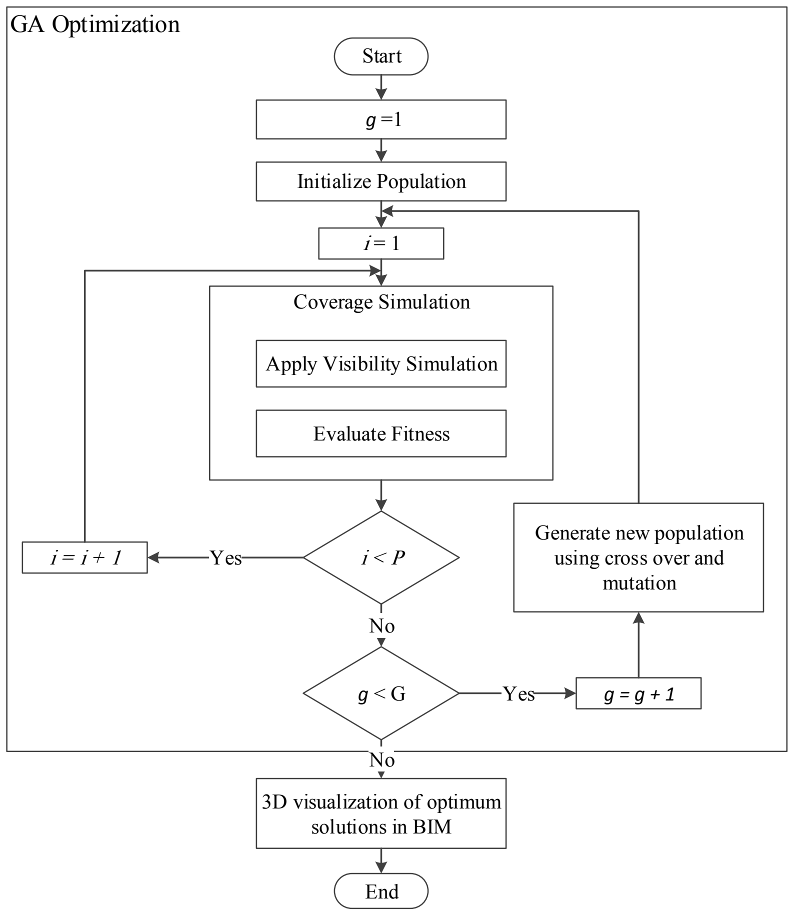


Figure 5-6 Proposed method

After the initialization process, the members of the population are sent to the coverage simulation process (i.e. 3D simulation process) in order to calculate the fitness (i.e. coverage) of each camera (*x*). This phase will be processed as the following: (1) generating equal cubic cells on the targeted surface (i.e. the floor) and assigning importance values (*IV*), which are selected by the specialist based on the location of the cells, as explained in (Albahri and Hammad, 2015); (2) adjusting the cameras' features (i.e. FOV, the camera height and clipping planes (far and near points)); (3) placing the cameras automatically based on the data of each gene given by the initialization process; and (4) calculating the fitness using Equations 5-1, 5-2, 5-3 and 5-4. To elaborate, Equation 5-1 is used to find the weighted number of cells in area a_i. The weighted covered cells (CCa_i) in area a_i can be calculated by the summation of all covered cells (C_{iv}) in an important area (a_i) as shown in Equation 5-2. Equation 5-3 is used to calculate the total coverage of camera (K_x). Then, Equation 5-4 is used to find the total coverage of all placed cameras. The output is sent to the GA to evaluate the first objective function, which aims to maximize the coverage of the cameras by applying Equations 5-4 to 5-9. The constraints are shown for the simple case of a rectangular shape room where the coordinate system is parallel to the walls.

$$W_{a_i} = IV_i \sum_{j=1}^{n} C_{ij}$$ Equation 5-1

$$CC_{a_i} = IV_i \sum_{v=1}^{n`} C_{iv}$$ Equation 5-2

$$Total\ coverage\ of\ camera\ K_x = \frac{\sum_{i=1}^{m} CC_{a_i}}{\sum_{i=1}^{m} W_{a_i}}$$ Equation 5-3

$$Maximize\ the\ total\ coverage\ of\ all\ cameras = \sum_{x=1}^{Q} K_x$$ Equation 5-4

Subject to: $0 \leq X \leq L$ Equation 5-5

$$H_{min} \leq Y \leq H_{max} \quad \text{Equation 5-6}$$

$$0 \leq Z \leq W \quad \text{Equation 5-7}$$

$$0 \leq Pan \leq 360° \quad \text{Equation 5-8}$$

$$0 \leq Tilt < 90° \quad \text{Equation 5-9}$$

where W_{a_i} is the weight of important area a_i, $C_{ij} = 1$ represents the cell j in important area i, and j = 1:n, IV_i is the importance value assigned to all cells in area i, and i = 1:m, $C_{iv} = 1$ is the covered cell v in area a_i and v = 1:n`, K_x is camera x and x = $1{:}\ Q$, CC_{a_i} is the weighted covered cells in area a_i, L is the maximum value of the length of the placement surface, H_{min} and H_{max} are the minimum and maximum values of the height of the camera on the wall, respectively, and W is the maximum value of the width of the placement surface.

The second objective function, which is minimizing the cost of cameras (Equation 5-10) is also calculated in the GA. Many features can affect the cost of surveillance cameras, such as the lens size, sensor type and the ability of night vision. The resolution of a camera is mainly affected by the characteristics of the camera sensor including the focal length, the pixel size and the projection type (Chen and Davis 2000). Based on a discussion with three experts in surveillance cameras, the most significant feature that can affect the cost of the cameras is the size of the lens, which has a direct impact on the FOV. A longer focal length results in a smaller FOV, and vice versa (Mansurov, 2015). The effect of the resolution is left to future work. For the sake of this research, each camera type is represented by its maximum FOV, which affects the coverage. The total cost is affected by the cost of the camera type (i.e. FOV) and the number of cameras.

$$\textit{Minimize the total cost of cameras} = \sum_{x=1}^{Q} N_x \qquad \text{Equation 5-10}$$

where N_x is the cost of camera x.

The values of the first and the second objective functions are saved to be ranked later (Pareto Front solutions). Next, the population of the next generation (g) is generated. The generation process includes applying the crossover and mutation operators. The crossover operator selects a pair of genes randomly and exchanges parts between them. It is important to note that in case of placing multiple cameras, the place of cutting is randomly selected by the crossover operator in the gene through each camera. For example, the place of cutting can be after the Z variable of every camera in the gene as shown in Figure 5-7.

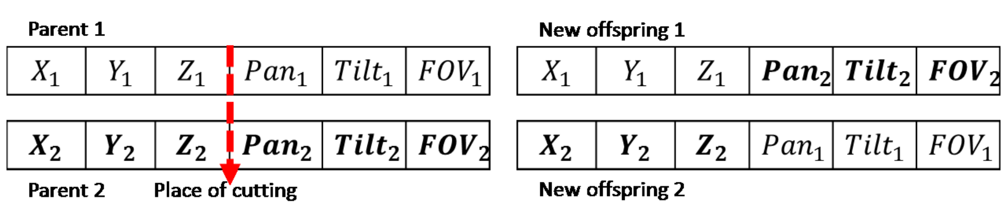


Figure 5-7 Crossover operator

The mutation operator randomly selects one variable of the gene and changes its value as shown in Figure 5-8. New generations will be created until the maximum number of generations (G) is reached.

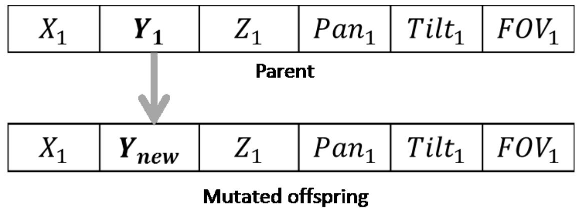


Figure 5-8 Mutation operator

The final results of the GA (Pareto front solutions) are sent to BIM tool in order to visualize the optimum cameras placement. Each model is representing a solution in the Pareto front with the following information: number, types, poses and FOV of cameras.

5.3 Implementation and case studies

A prototype system was developed to validate the usability of the method using BIM, Unity3D game engine (Unity, 2015) and Matlab (Matlab, 2015). A case study was conducted on the ground floor of the Engineering and Visual Art Building in Concordia University. The similar BIM model of chapter 4 was used in this case study. The floor has some physical constraints such as columns and stairs that can affect the cameras coverage. The BIM model is sent to the Unity3D simulation module using FBX format for facilitating the computational processes, which were developed using C# programming language.

The floor of the model was divided into a number of important areas in Unity3D. Unlike the previous chapter, the importance values were assigned as the following: (1) value of 1 was assigned to cells which cover general areas such as corridors; (2) value of 3 was assigned to cells which cover stairs and elevators' halls; and (3) value of 6 was assigned to cells which cover entrances and escalators coming from the underground metro station.

The important areas were covered by cells in a way that covers the upper limit of the average human height.

As previously highlighted in Chapter 4, the size of the cell is an important factor that can affect the coverage calculation accuracy. The smaller cell size will increase the number of cells and will increase the computational time but will provide more accurate results, and vice versa. By covering the monitored areas with cells, it will be possible to determine the volume of important areas that is not covered by the placed cameras. These cells are assigned by the user with different *IV* based on their locations. Cells with an edge of 0.7 m were selected as the suitable cell size. Also, a grid system was developed in order to facilitate the camera placement process.

The case study is divided into two parts. The first part aims to optimize the placement of a single camera, while the second part aims to find the optimum placement of multiple cameras. Figure 5-9 shows the implementation flowchart of the proposed method. The process includes integration between the GA module and the simulation module. This integration is useful in assessing each solution (i.e. gene) generated by the GA. At the time being, the Unity3D is linked with the GA tool using text files. In order to apply the simulation-based optimization process, the following steps were implemented:

(1) Obtaining the valid locations of the cameras placement from BIM model. This includes specifying the decision variables (i.e. *X*, *Y*, *Z*, Pan, Tilt angles and FOV) that will be used in the GA module. These decision variables are defined with ranges (i.e. minimum and maximum values), which will be used for defining the search space.

(2) Determining the number of solutions and generations using the interface of the GA module. This module uses the PICEA GA toolbox developed by De Freitas (2012) within Matlab to conduct the optimization process. The GA toolbox requires determining the ranges for each parameter that affects the GA process.

(3) Initializing the population of the first generation and saving the results in a text file. The text file is exported to the simulation module. A sample of the text files (in case of multiple cameras) is given in Appendix G.

(4) The simulation module receives the text file and conducts the camera placement based on the decision variables in order to calculate the first objective function (i.e. the camera coverage).

(5) The coverage objective function of the selected camera is calculated based on the visibility analysis.

(6) The results of the camera coverage in each solution are saved into a text file and returned to the GA module. A sample of the text file is given in Appendix H.

(7) The GA evaluates the results. In case of multiple cameras types, the GA calculates the cost objective function, saves the results and initiates the next generation.

(8) The GA applies the crossover and mutation processes to generate more suitable solutions. This process ends when the maximum number of generations is reached. The code of applying the crossover and mutation processes is given in Appendix D.

(9) The results are ranked in the GA in order to find the optimum solutions. A trade-off process is conducted by selecting the gene that has the least cost and low coverage rate.

The process is continued by searching for genes with higher cost and as much higher coverage as possible.The final Pareto set includes the sets of cameras placement with the best fitness values.

(10) The optimum solutions are sent to the simulation module to be visualized.

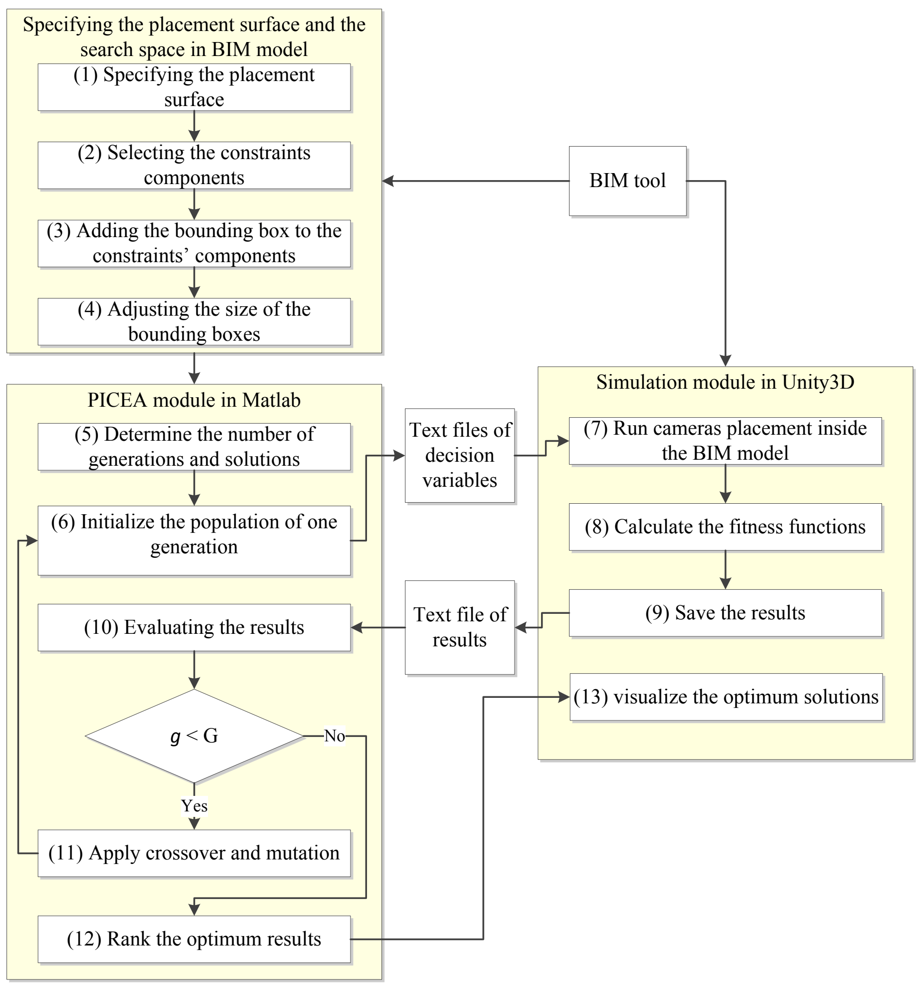


Figure 5-9 Implementation flowchart

5.3.1 Optimizing the placement of a single camera

This section aims to find the optimum placement for a single fixed camera using BIM, without considering its cost or type. As previously explained, the first step is to obtain the valid locations of the camera placement using Autodesk Revit, as shown in Figure 5-10. This is done by excluding the invalid locations, which are inside the yellow buffers. The buffer size is assumed equal to 50 cm in this case study. The constraints buffers surround the locations of some columns and HVAC components on the ceiling surface. Positions that are not within the constraints buffers are considered as the valid locations that are not affected by the constraints. These valid locations are sent to Matlab in order to be later used in generating the placement solutions.

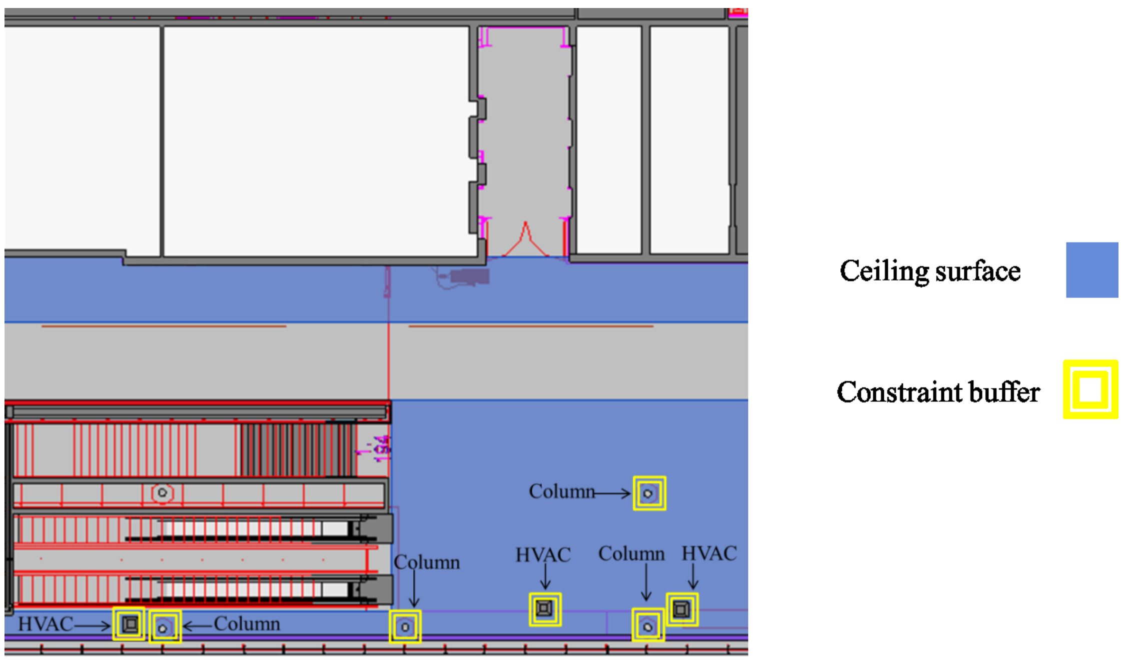


Figure 5-10 Obtaining the valid camera locations using Autodesk Revit

The BIM model is sent to Unity3D in order to facilitate the computational processes. The optimum placement is where the camera can offer the maximum coverage for areas a_5-a_8 as shown in Figure 5-11. These areas were covered by cells that have different importance values (i.e. value of 1 for the lowest importance, 3 for the medium importance and 6 for the highest importance). Furthermore, the placement surface has the dimensions of 25 m on the Z axis and 14 m on the X axis at a height of $Y = 5$ m. The placement surface (shown in green) is divided by a grid that can define the camera placement with the spacing of 25 cm. This grid results in 56×100 locations of the camera.

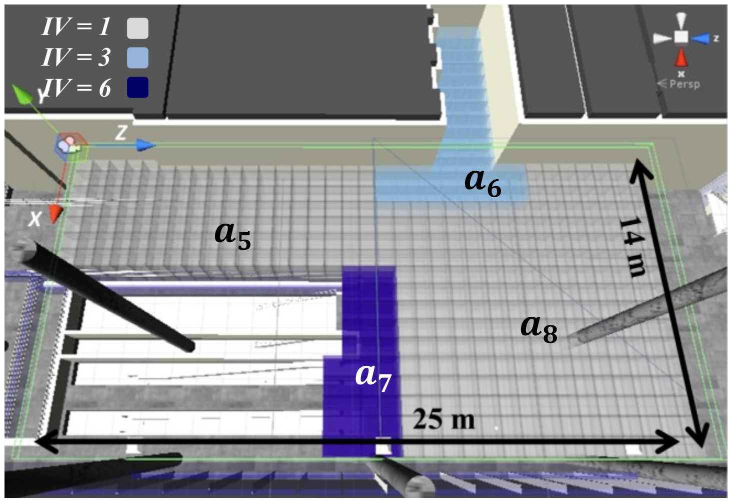
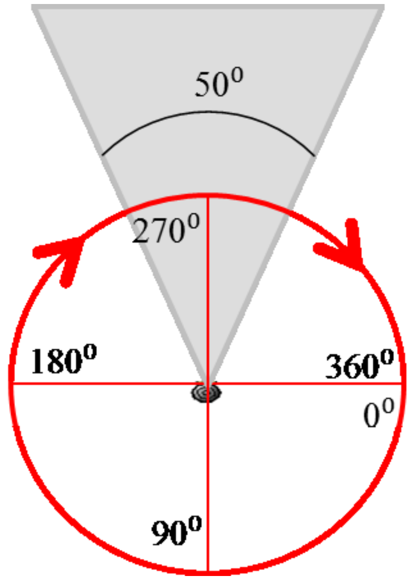
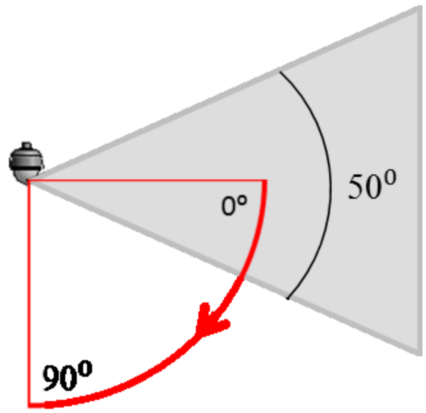


Figure 5-11 Monitored areas with importance values and the placement surface

As previously discussed, the camera placement is affected by the number of possible camera poses. Each pose includes the location of the camera and the pan and tilt angles, which represent the orientation of the camera, as shown in Figures 5-12 (a) and (b). The horizontal and vertical FOV angles of the camera are fixed to 50°. Table 5-1 summarizes the ranges of the decision variables related to this case study along with their minimum and maximum values and the step. Also, it is important to mention that the total number of combinations of the search space in this case study is 56×100×360×90 = 181,440,000. It took approximately 5.25 hours to solve this problem using the proposed method on a computer with AMD A4 QUAD-CORE-5000 APU processor and 8 GB of Random Access Memory (RAM).



(a) Plan view showing the range of camera pan angle



(b) Section view showing the range of camera tilt angle

Figure 5-12 Ranges of pan and tilt angles

The maximum number of generations and the population size of the GA are selected as 65 and 250, respectively. The crossover and mutation rates used in this case study are 90% and 5%, respectively.

Table 5-1 Ranges of decision variables

Decision variables	*Minimum*	*Maximum*	*Step*
X (cm)	0	1,400	25
Y (cm)	500	500	N.A.
Z (cm)	0	2,500	25
Pan (°)	0	360	1
Tilt (°)	0	90	1

Table 5-2 shows the best 10 solutions with their coverage results. Figure 5-13 shows some of these solutions visualized using Unity3D. For example, in solution 10, the camera placement was at X = 100 cm, Y = 500 cm, Z = 2,500 cm, pan angle = 205°, and tilt angle = 61°. This solution offered camera coverage of 65%. Solution 1 has the best coverage result of 90% by fixing the camera placement at X = 1,350 cm, Y = 500 cm, Z = 1,600 cm, pan angle = 115° and tilt angle = 62°. Figure 5-14 shows the global fitness, best fitness and the mean fitness curves.

Table 5-2 Best 10 solutions of the optimization process

Solution#	*X (cm)*	*Y (cm)*	*Z (cm)*	*Pan angle (°)*	*Tilt angle (°)*	*Coverage (%)*
1	1,350	500	1,600	115	62	90
2	1,250	500	1,550	120	59	89
3	1,225	500	1,600	102	65	88
4	1,400	500	1,500	88	59	86
5	1,200	500	1,500	120	57	84
6	100	500	1,100	200	60	82
7	50	500	100	306	52	78
8	100	500	150	116	80	74
9	50	500	1,500	170	48	69
10	100	500	2,500	205	61	65

The global curve represents the maximum values the GA achieved during the optimization process. The best curve represents the maximum value achieved in each generation. The mean curve represents the mean value achieved in each generation. The global fitness curve shows an increase from generation 9 to generation 65. The global fitness value is 90%, which existed from generation 53 to generation 65.

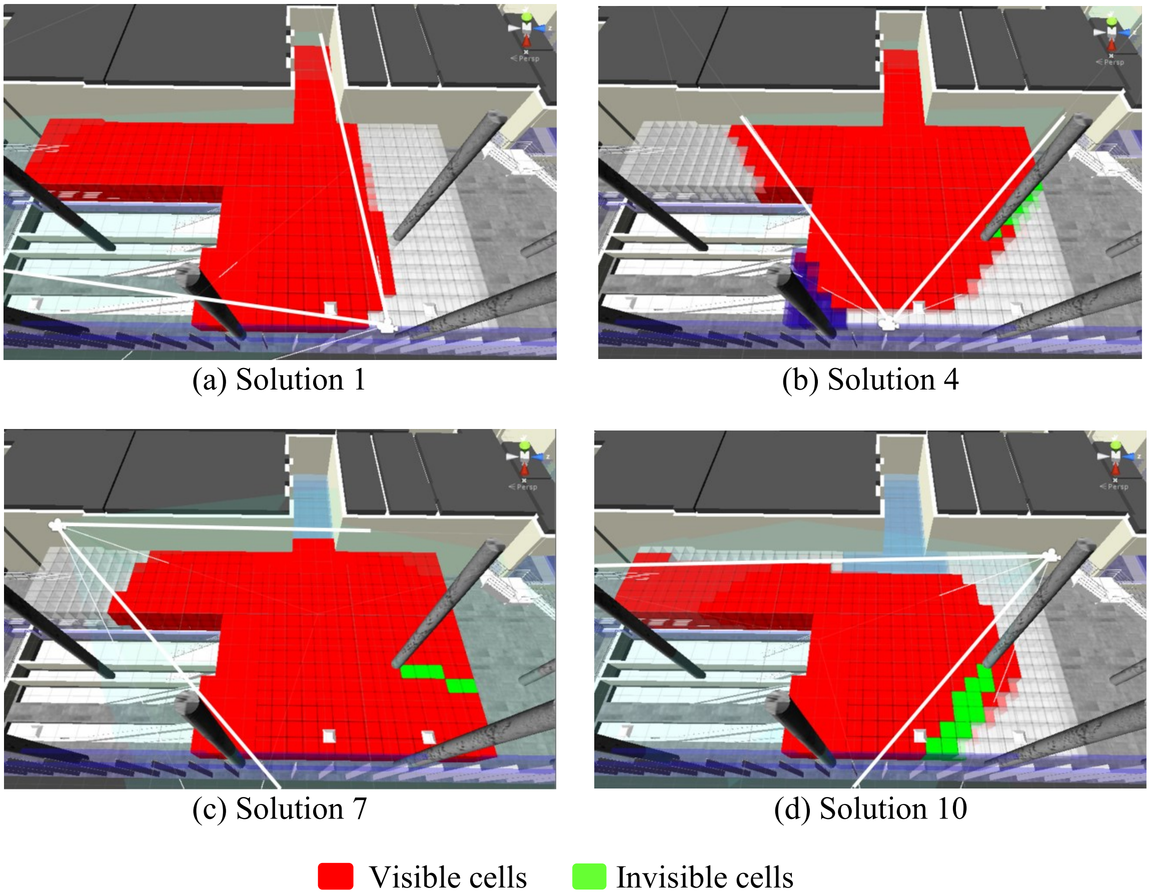


Figure 5-13 Visibility simulation of some solutions

The best fitness curve shows a number of fluctuations throughout different generations. For example, this curve shows a fluctuation ranging between 59% to 78% coverage from generation 8 to generation 27. The curve gained coverage of 82% in generation 28, and kept increasing till it reached 90% coverage in generation 53. It is important to note that the best results were produced by placing the camera near the edge of the placement surface with the appropriate pan and tilt angles.

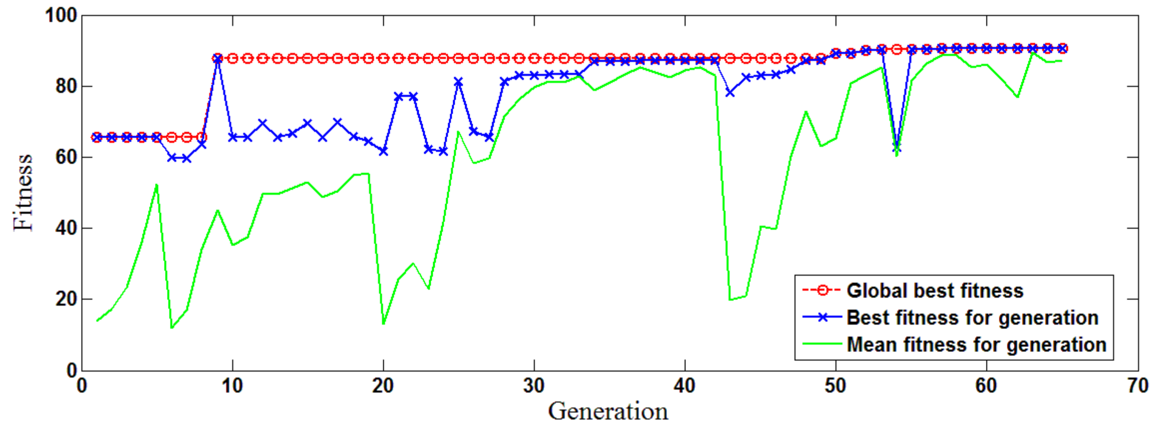


Figure 5-14 Behaviour of fitness values throughout generations

5.3.2 Optimizing the placement of multiple cameras of different types

This case study aims to optimize the placement of a number of cameras with multiple camera types in a large space that cannot be efficiently covered using a single camera. The optimization problem aims to maximize the coverage and to minimize the total cost of cameras. Four types of cameras were selected (Axis Communication, 2015), to be used in this case study with FOV values of 60°, 80°, 92° and 100° as shown in Table 5-3. The monitored area includes the important areas a_2 to a_8 as shown in Figure 5-15. The placement surface has the dimensions 14 m in the *X* axis and 49 m in the *Z* axis.

Table 5-3 FOV and cost of the selected camera types

Camera type	Cost ($)	FOV(°)
1	293	60
2	419	80
3	700	92
4	930	100

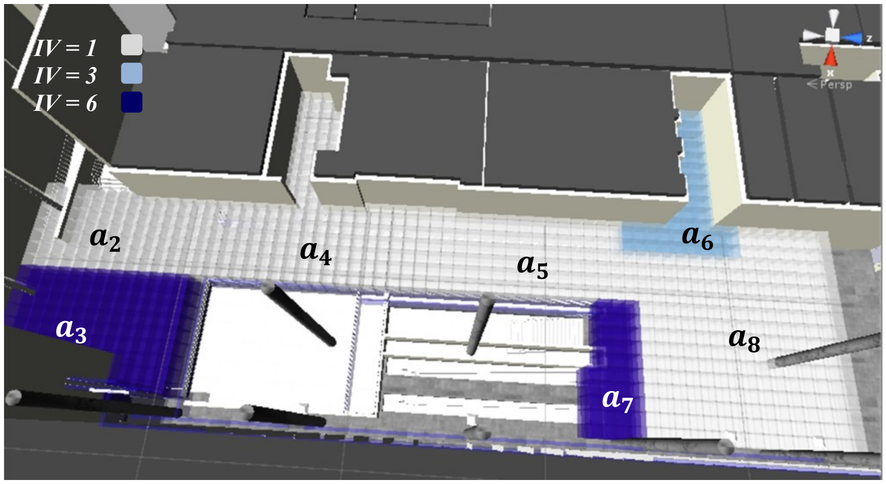


Figure 5-15 Important areas from a_2 to a_8

Values of maximum number of generations and solutions, and mutation and crossover rates are similar to the previous case study. Figure 5-16 shows the dominated and non-dominated Pareto solutions, with the convergence of the last four generations, and highlights six Pareto solutions in the final generation.

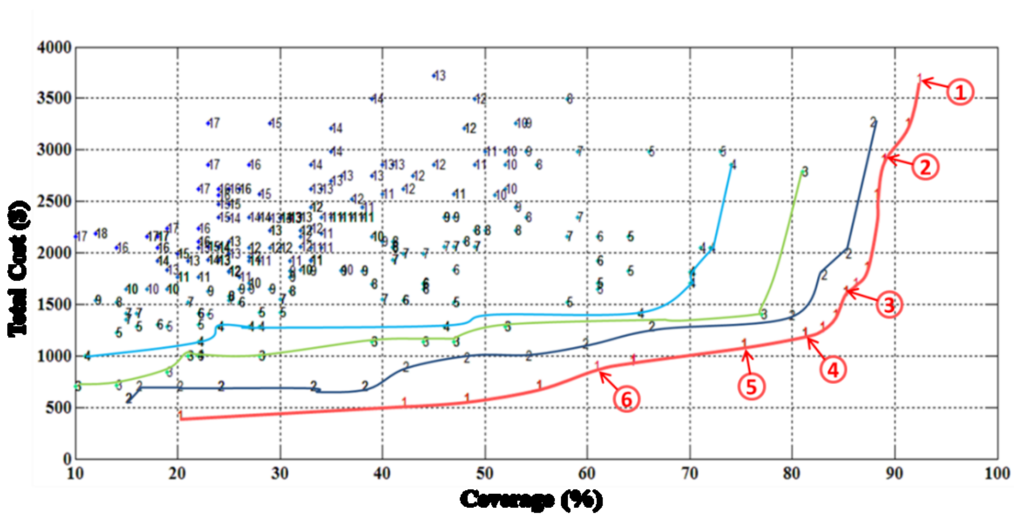


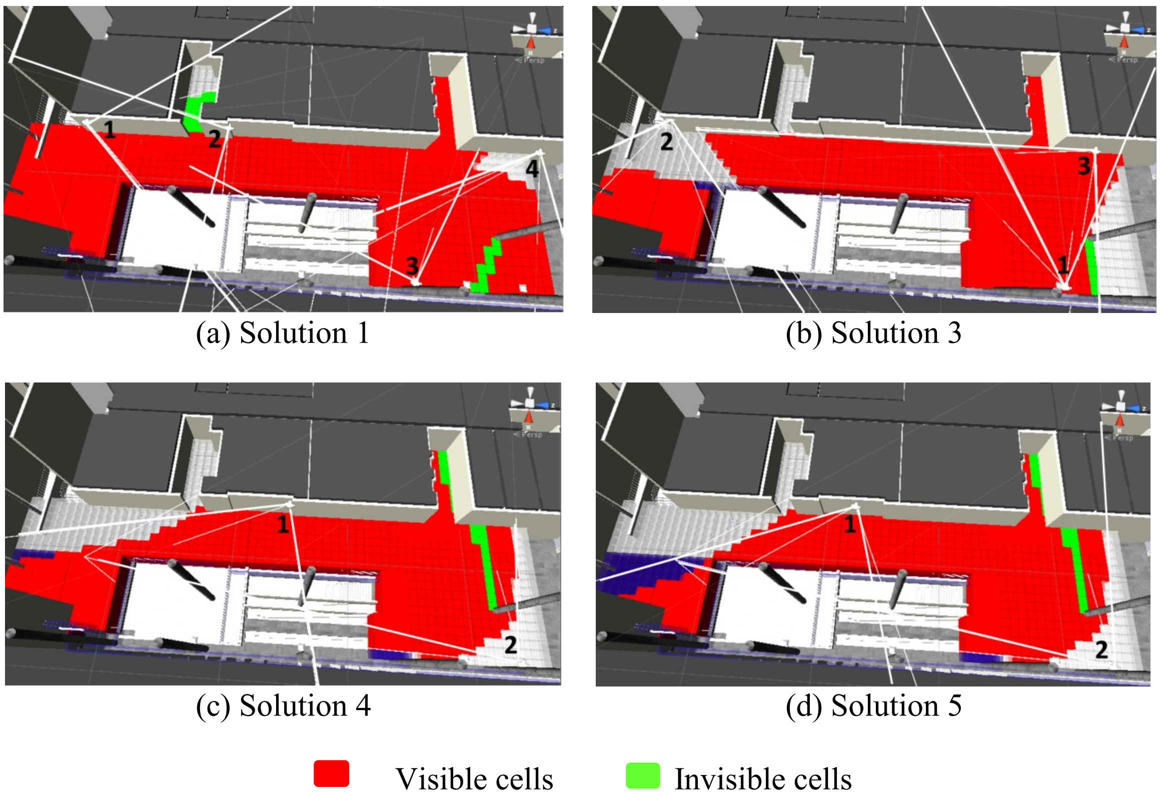
Figure 5-16 Dominated and non-dominated Pareto solutions

The details about these solutions are shown in Table 5-4. It can be noticed that solution 1 combining four cameras of the same type (FOV= 100°) can ensure the maximum coverage of 92% among all other Pareto front solutions. However, this solution is the most costly ($3,720).

Table 5-4 Examples of the non-dominated Pareto solutions

Solution #	*Number of Cameras*	*Camera parameters*							*Coverage (%)*	*Total cost ($)*
1	4	Camera ID	Camera type	X (cm)	Y (cm)	Z (cm)	Pan (°)	Tilt (°)	92	3,720
		1	4	50	500	800	5	55		
		2	4	0	500	2,050	188	68		
		3	4	1,350	500	3,350	98	65		
		4	4	50	500	4,900	254	41		
2	4	Camera ID	Camera type	X (cm)	Y (cm)	Z (cm)	Pan (°)	Tilt (°)	89	2,853
		1	1	50	500	700	300	38		
		2	4	100	500	2,250	220	44		
		3	4	1,400	500	2,200	100	73		
		4	3	1,350	500	4,800	110	67		
3	3	Camera ID	Camera type	X (cm)	Y (cm)	Z (cm)	Pan (°)	Tilt (°)	85	1,693
		1	1	1,350	500	4,050	90	71		
		2	3	0	500	750	260	55		
		3	3	50	500	4,650	202	41		
4	2	Camera ID	Camera type	X (cm)	Y (cm)	Z (cm)	Pan (°)	Tilt (°)	81	1,223
		1	4	0	500	2,450	258	48		
		2	1	1,350	500	4,750	111	34		
5	2	Camera ID	Camera type	X (cm)	Y (cm)	Z (cm)	Pan (°)	Tilt (°)	76	1,119
		1	3	50	500	2,250	261	55		
		2	2	1,400	500	4,750	108	66		
6	3	Camera ID	Camera type	X (cm)	Y (cm)	Z (cm)	Pan (°)	Tilt (°)	61	879
		1	1	1,350	500	4,700	120	50		
		2	1	50	500	4,850	250	48		
		3	1	1,350	500	2,550	102	45		

Figure 5-17 shows the simulation results of solutions 1, 3, 4 and 5. Each solution includes a number of cameras attached with their IDs, which represent the camera types as explained in Table 5-4. Solutions 3 and 4 provide similar coverage with different cost options. To elaborate, solution 3 offers 85% coverage with a cost of $1,693 and a combination of two types of cameras (i.e. two cameras with FOV= 92° and one camera with FOV= 60°). While solution 4 provides the coverage of 81% with a lower cost of $1,223 and a combination of two types of cameras (i.e. one camera with FOV= 60° and one camera with FOV= 100°).



(a) Solution 1 (b) Solution 3

(c) Solution 4 (d) Solution 5

Figure 5-17 Four of the non-dominated Pareto solutions attached with their cameras' ID

5.4 Summary and conclusions

This chapter presented a simulation-based optimization method for selecting the camera types, number and placement in a building using BIM model. The optimal placement of cameras aims to maximize the camera coverage and to minimize the cost. In addition, BIM tool is used to define the constraints of the placement surface for the purpose of specifying the valid locations of the camera placement. The method includes the following steps: (1) specifying the valid locations, (2) generating solutions, (3) coverage simulation, (4) evaluating the results, and (5) visualizing the optimum results. The feasibility of the method was demonstrated using two case studies: (1) optimizing the placement of a single camera; and (2) optimizing the placement of multiple camera types. The results of the case studies showed that the maximum coverage can be achieved by placing the cameras near the edge of the placement surface with the appropriate pan and tilt angles. Also, the results showed the ability to increase the coverage by adding more cameras with different FOV angles. The maximum coverage was 92% by adding four cameras each with FOV = 100°. The main contributions of the chapter are: (1) proposing a method of specifying the valid locations of cameras using BIM tool; and (2) proposing a method that can find near optimum for the number, type and placement of cameras inside building taking into consideration the existence of different geometrical constraints on the placement surface. The proposed method can help individuals who are responsible of the camera installation to efficiently determine the optimum types, number and placement of cameras needed to monitor spaces in buildings.

CHAPTER 6: SIMULATION-BASED OPTIMIZATION OF PTZ CAMERA PLACEMENT AND MOVEMENT PLAN IN BUILDINGS USING BIM

6.1 Introduction

As earlier research focused on optimizing the placement of fixed cameras, further research is needed regarding the placement of PTZ cameras. PTZ cameras are mostly used in large public buildings such as airports and shopping malls (Honovich, 2009). These cameras are programmed to rotate among several targets according to a schedule. It is necessary to understand the difference between fixed cameras and PTZ cameras. The purpose of using fixed cameras is to ensure the constant coverage of the monitored area considering that the orientation (i.e. pan and tilt angles) of the camera is not changing during the monitoring process. On the other hand, PTZ cameras are used as an addition to fixed cameras in order to track dynamic activities (Krahnstoever et al., 2008). During this tracking, the orientation of the PTZ camera is changing within the camera cycle path (*CP*). The *CP* is composed of a number of phases, which represent the stopping places during specific times and the time to move from one phase to another. The movement among phases can be manually controlled by the security personnel or automatically by storing a suitable movement plan in the camera system. However, finding the optimum movement plan is not an easy task. The optimum placement of PTZ cameras should guarantee adequate coverage when the camera is stopping to detect an activity over a suitable time, which is called the *phase coverage-time* in this chapter. Recent research associated with PTZ cameras focused on three lines of research: (1) enhancing the detection, tracking and recognition abilities of PTZ cameras; (2) optimizing the handoff planning between PTZ cameras; and (3) optimizing the placement of PTZ cameras using

2D methods. So far, there is no scientific method that can find the optimum placement of a PTZ camera considering its rotation and coverage-time for every activity. Also, one main issue affecting the coverage of cameras is in the case where the number of activities (i.e. targets) exceeds the number of placed cameras (Krahnstoever et al., 2008).

The objective of this chapter is to find the near-optimum placement and phases for a single PTZ camera in a way that ensures the maximum *CP* coverage. The *CP* coverage is the average coverage of the PTZ camera during its cycle path time. Also, this chapter uses BIM tool to specify the placement surface of the PTZ camera. A simulation-based optimization method is developed using BIM tool and GA in order to solve the optimization problem. This research proposes specifying a number of important areas that cover the dynamic activities and are covered with cells. The proposed method includes three major steps: (1) specifying the placement area in the BIM tool; (2) applying GA optimization; and (3) applying the visibility simulation process. The optimization includes a single objective function, which aims to maximize the *CP* coverage of the monitored area. The rest of this chapter includes the following. Section 6.2 describes the proposed method. Section 6.3 provides the implementation and a case study. Section 6.4 includes the summary and conclusions.

6.2 Proposed method

The proposed method includes three main modules. The first module starts by specifying the search space using BIM. This is done by excluding the locations of the obstructing elements and specifying the valid locations of the camera placement. The second module focuses on the GA optimization process. The third module represents the visibility simulation process, which includes the following two processes: (1) preparing the 3D simulation environment, which is done one time; and (2) calculating the *CP* coverage of the PTZ camera.

6.2.1 Specifying the search space using BIM

As explained in Chapter 5, BIM tool is used to automatically specify the valid locations (i.e. search space) of the camera on the placement surface. This is because of the ability of BIM tool to identify the geometrical and non-geometrical attributes of the constraints on the placement surface. Using the constraints buffers, the points outside these buffers are considered as the valid locations of the PTZ camera placement.

6.2.2 GA optimization pre-processing

6.2.2.1 Cycle path of PTZ cameras

As explained in the Section 6-1, the orientation of the PTZ camera can change to cover the dynamic activities around the camera *CP* and within the *CP* time (*T*). The *CP* is composed of a number of phases (p_k) (i.e. stopping places) of durations T_{p_k}, separated by travel times (T_{t_k}). *T* is the summation of T_{p_k} and T_{t_k} of all phases, as shown in Equation 6-1.

$$T = \sum_{k=1}^{K} (T_{p_k} + T_{t_k}) \qquad \text{Equation 6-1}$$

where T_{p_k} is the time spent in phase p_k, $T_{t_k}$is the travel time between p_k and p_{k+1}, and K is the total number of phases. The values of T_{p_k} are randomly generated by the optimization as will be explained in the next section. The value of *T* is selected based on the size of the monitored area.

Every phase (p_k) corresponds to the coverage of the camera's FOV when it is stopping for a specified time (T_{p_k}). For simplicity, Figure 6-1 shows a 2D example of a *CP* in the pan direction with a full 360° rotation. In order to optimize the PTZ camera placement, the *CP* coverage should be maximized. It should be noted that this research only considers the coverage during T_{p_k}, while fixing the rotation speed during T_{t_k} to the maximum value allowed by the camera.

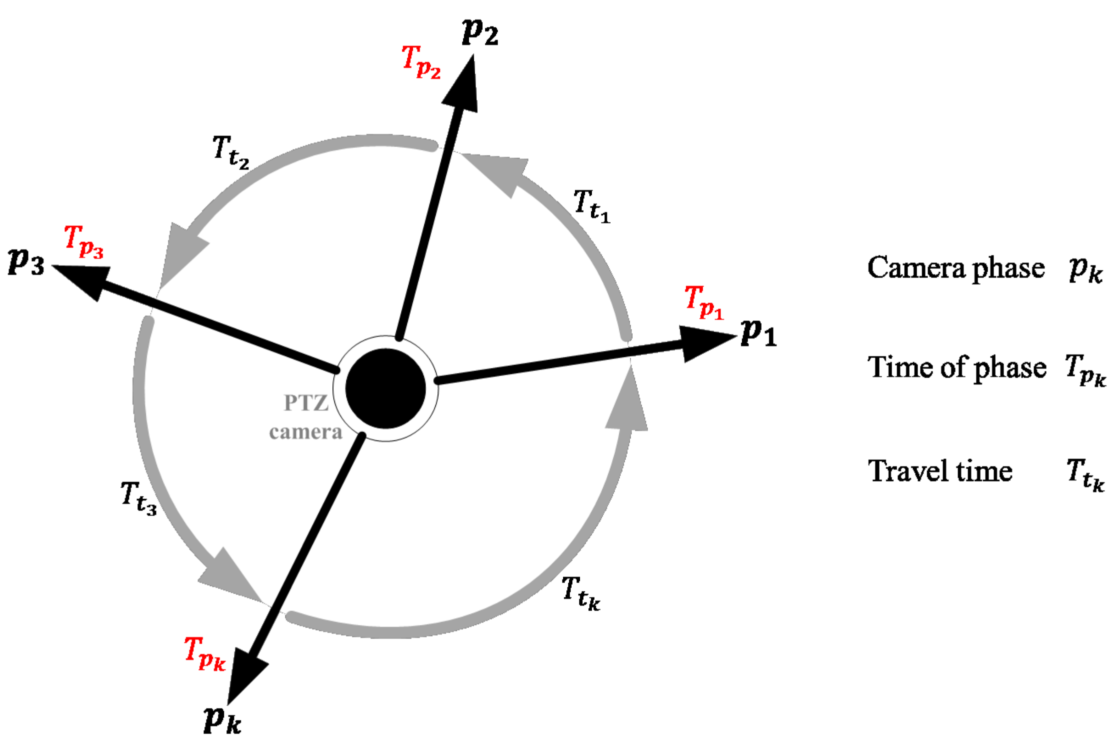
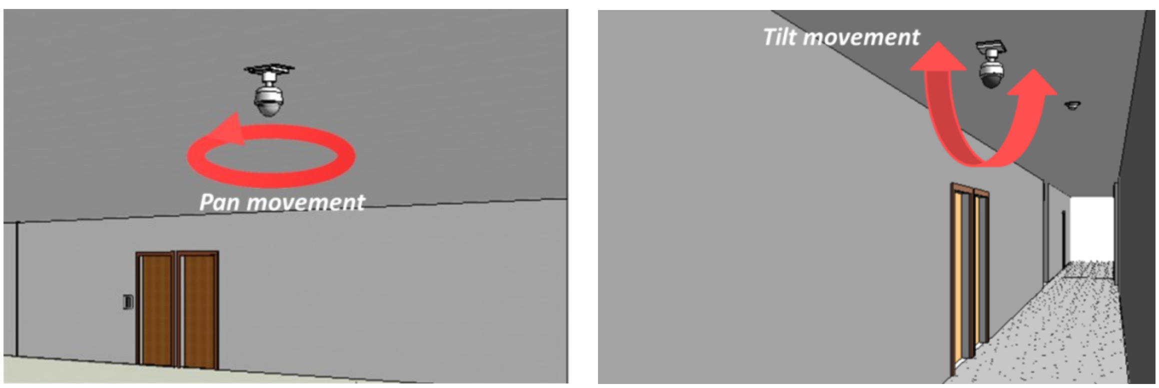


Figure 6-1 Example of the cycle path of PTZ camera

6.2.2.2 Determining the number and ranges of phases

It is important to ensure the logical sequence of phases and to reduce the overlapping between phases so that it does not exceed a predefined percentage. This can be achieved by specifying the dominant movement for the PTZ camera. The dominant movement is the one that dominates the camera direction within most of its *CP*. This movement is affected by the location of the PTZ camera and the surrounding important areas. The dominant movement might be in the pan direction in case the camera is located in a wide area (e.g. a hall) and the important areas surround the camera from all directions as shown in Figure 6-2 (a). In this case, the tilt movement works as a supporting (secondary) movement. Another case is when the camera is located in a narrow area (e.g. a corridor) and the important areas are distributed along this area, which makes the tilt direction the dominating direction of the camera movement, as shown in Figure 6-2 (b). This will leave the pan movement to work as the supporting movement.



(a) Pan direction (b) Tilt direction

Figure 6-2 Dominant movements of the PTZ camera

After determining the dominant movement of the PTZ camera, the number of phases can be calculated using Equation 6-2.

$$K = \frac{Angle\ of\ dominant\ movement}{FOV\ angle} \qquad \text{Equation 6-2}$$

where K is the number of phases.

The next step is to determine the separation angle, which can ensure that the overlapping between phases is not exceeding an acceptable percentage. The separation angle (α) is calculated using Equation 6-3.

$$\alpha = FOV\ angle\ \times Acceptable\ overlapping\ percentage \qquad \text{Equation 6-3}$$

The camera orientation in each phase is restricted within a specific range, which is determined by α, as shown in Equation 6-4. O_{min} and O_{max} can be calculated using Equations 6-5 and 6-6, respectively.

$$O_{min}{}^{k} \le camera\ orientaion < O_{max}{}^{k} \qquad \text{Equation 6-4}$$

$$O_{min}{}^{k} = FOV\ angle \times (k-1) \qquad \text{Equation 6-5}$$

$$O_{max}{}^{k} = O_{min}{}^{k} + FOV\ angle - \alpha \qquad \text{Equation 6-6}$$

where O_{min} and O_{max} are the minimum and the maximum angles of the phase.

As an example, if the dominant movement is in the pan direction (i.e. the angle of the dominant movement is 360°), the FOV of the camera is 90° and the acceptable overlapping percentage is 50%, then the total number of phases is four and α = 45°. The camera orientation in each phase is randomly selected within the restricted range of the phase, as shown in Figure 6-3.

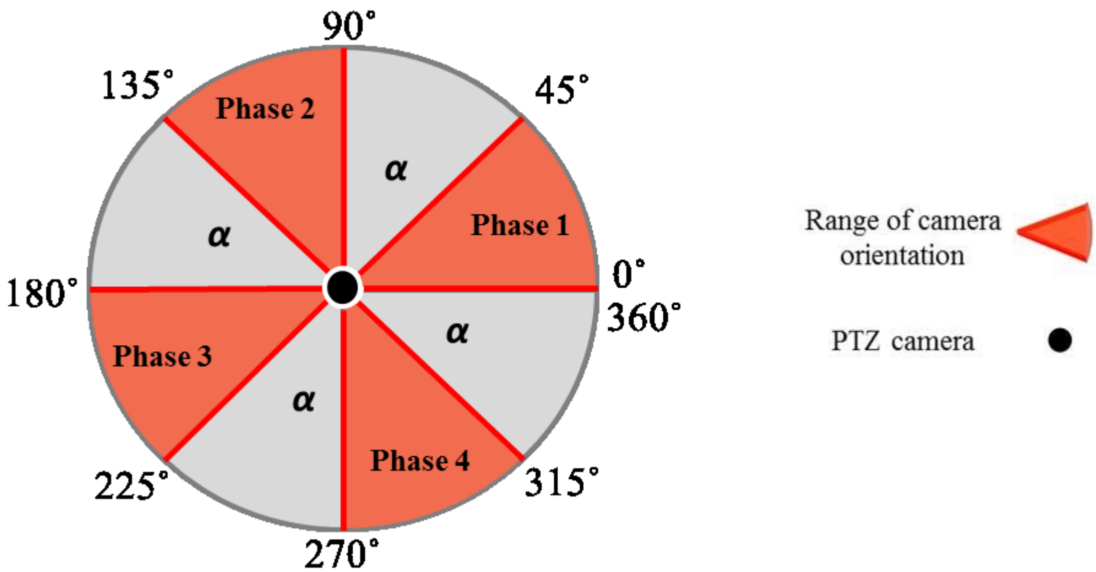


Figure 6-3 Selecting the suitable camera orientation for each phase in the case of the pan dominating movement (FOV = 90° and α = 45°)

It is important to set the suitable ranges for T_{p_k} taking into consideration Equation 6-1. The minimum value of ${T_{p_k}}^{max}$ is calculated using Equation 6-7. For example, if T = 80 sec. and the number of phases = 4, then the maximum value of T_{p_k} should be equal or greater than 20 sec.

$$({T_{p_k}}^{max})_{min} \geq \frac{T}{K} \qquad \text{Equation 6-7}$$

Figure 6-4 shows the algorithm for calculating T_{p_k} and T_{t_k}. The algorithm starts by determining K, as explained in Equation 6-2. T_{t_k} values are calculated based on the travel angular distance and the maximum speed of the camera. Then, the total phases' times (U) is calculated as shown in Equation 6-8.

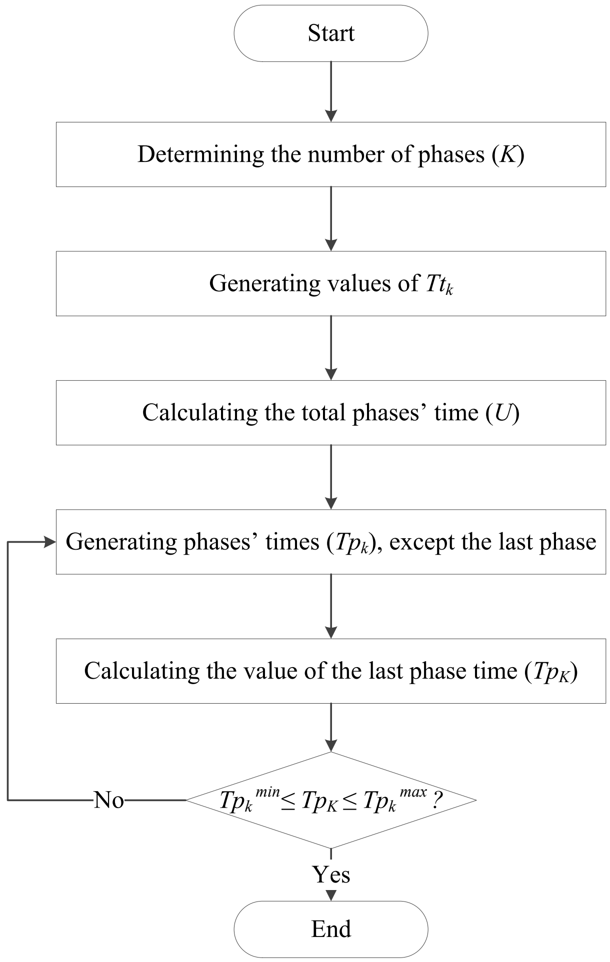


Figure 6-4 Calculating the phases' times

Random T_{p_k} values are then generated, respecting Equation 6-9, except the last phase time (T_{p_K}), which is calculated using Equation 6-10. If the conditions are not satisfied, the GA will have to generate new random values of T_{p_k}.

$$U = T - \sum_{k=1}^{K} T_{t_k} \qquad \text{Equation 6-8}$$

$$T_{p_k}{}^{min} \leq T_{p_K} \leq T_{p_k}{}^{max} \qquad \text{Equation 6-9}$$

$$T_{p_K} = U - \sum_{k=1}^{K-1} T_{p_k} \qquad \text{Equation 6-10}$$

where $T_{p_k}{}^{min}$ and $T_{p_k}{}^{max}$ are the minimum and maximum values of the phase time, respectively, U is the total phases' times, and T_{p_K} is the time of the last phase.

6.2.2.3 Generating the solutions for optimizing the PTZ camera placement

After obtaining the valid camera locations from BIM tool, this step implements the GA optimization for the PTZ camera placement as shown in the left side of Figure 6-5. This starts by initializing the first generation (*g*), which has a specific population size (*P*). This generation includes random solutions of the camera placements and *CP*. Each solution includes a specific number of phases. Each phase (p_k) includes: (1) the location of the camera on the placement surface (*X, Y, Z* coordinates); (2) pan and tilt angles; and (3) the phases' times (T_{p_k}). Each solution is sent to the visibility simulation process in order to calculate its *CP* coverage.

6.2.3 Visibility simulation process

This step is divided into two parts as the following: (1) preparing the 3D simulation environment; and (2) calculating the *CP* coverage. The first part includes developing a grid system on the placement surface. The grid system can facilitate the camera placements, which are generated by the GA. Also, this part includes generating a number of 3D cubic cells on the monitored areas that include the dynamic activities. The cells should cover the upper limit of the average human height. As explained in Chapter 4, each cell is assigned with an importance value based on its location. This part ends by selecting the camera and attaching it to the placement surface in order to be ready for the simulation.

After preparing the 3D simulation environment, the part of calculating the cycle path coverage is conducted based on the solutions received by the GA, as shown in the right side of Figure 6-5. To calculate the *CP* coverage, the following steps are conducted in the visibility simulation process: (1) Placing the PTZ camera based on the given solution; (2) Detecting the visible cells located in all areas a_i within phase p_k. The visibility simulation process uses ray tracing for the purpose of detecting the importance values of cells. During the visibility process, the ray line (*R*) will read the *IV* of the cell (C_{ij}) and consider the T_{p_k} given by the solution in each phase p_k; (3) Calculating the coverage-time in phase p_k using Equation 6-11 for all phases (*K*); and (4) Calculating the *CP* coverage of the PTZ camera by using Equation 6-12. (5) Conducting the previous steps on the next candidate solution until finishing from all solutions in the current population; and (6) Sending the results back to the GA.

$$CT_{p_k} = T_{p_k} \sum_{i=1}^{m} \sum_{v=1}^{n`} (C_{iv} \times IV_i) \qquad \text{Equation 6-11}$$

$$C_{CP} = \frac{\sum_{k=1}^{K} CT_{p_k}}{T \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} \times IV_i} \qquad \text{Equation 6-12}$$

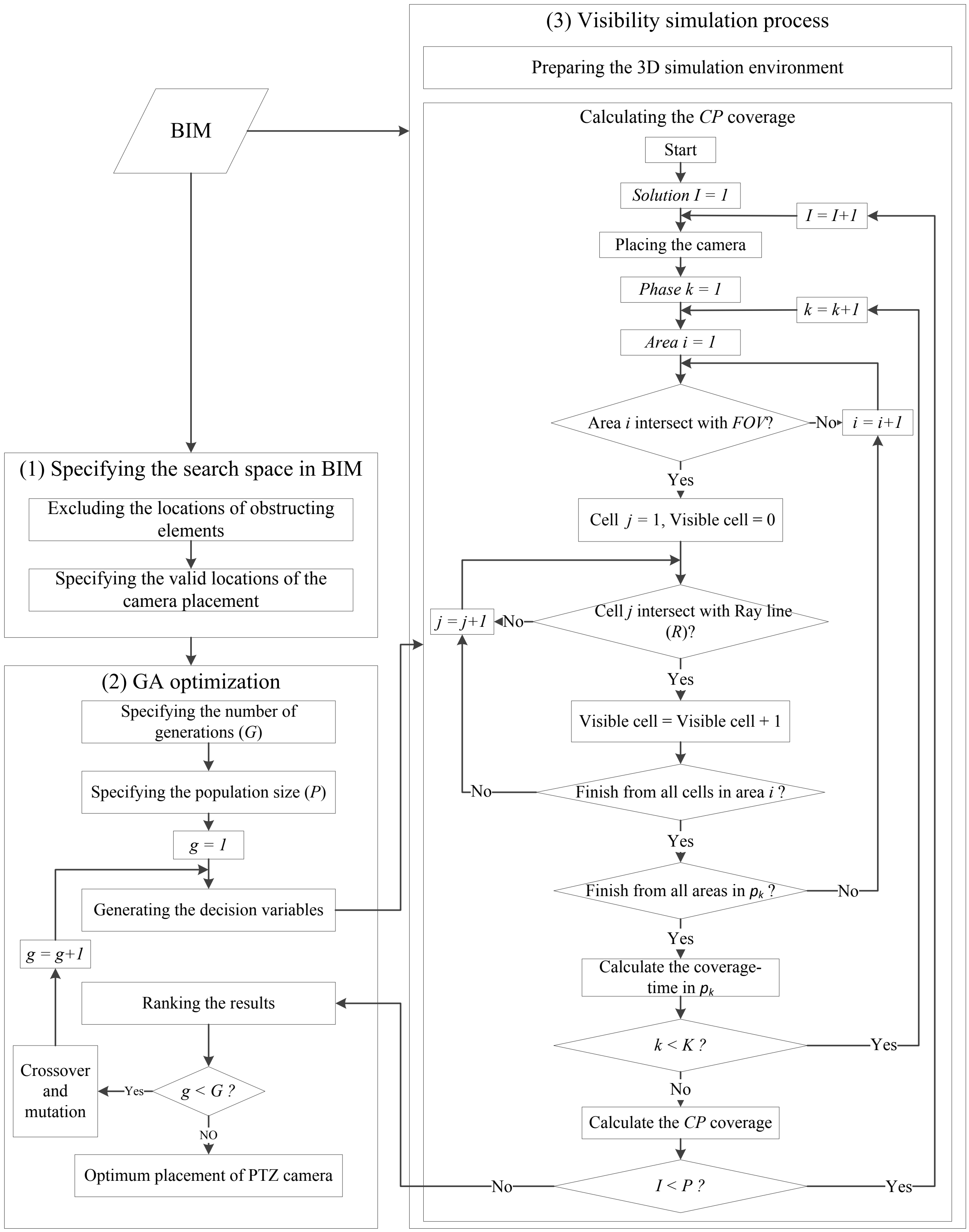


Figure 6-5 Method of optimizing the placement of the PTZ camera

where: CT_{p_k} is the coverage-time in phase p_k , T_{p_k} is the phase time of p_k, C_{iv} is the covered cell v in area a_i and v = 1:n`, i = 1:m, IV_i is the importance value of cell C_{iv} in area a_i, C_{CP} is the CP coverage, K is the total number of phases, $k = 1 : K$, T is the total time of the cycle path, C_{ij} is the covered cells in area i, j = 1:n, and IV_i is the importance value of cell C_{ij} in area a_i.

6.2.4 Using the results of simulation in the GA

The results of simulating every generation are ranked in the GA to find the maximum values of *CP* coverage and used to generate other populations considering Equations 6-13 to 6-20.

$$Maximize\ C_{CP} = \frac{\sum_{k=1}^{K} CT_{p_k}}{T \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} \times IV_i} \quad \text{Equation 6-13}$$

Subject to:

$$0 \leq X \leq L \quad \text{Equation 6-14}$$

$$H_{min} \leq Y \leq H_{max} \quad \text{Equation 6-15}$$

$$0 \leq Z \leq W \quad \text{Equation 6-16}$$

If the pan direction is dominating the camera movement

$$O_{min}{}^{k} \leq Pan\ angle < O_{max}{}^{k} \quad \text{Equation 6-17}$$

$$0° \leq Tilt\ angle \leq 90° \quad \text{Equation 6-18}$$

If the tilt direction is dominating the camera movement

$$O_{min}{}^{k} \leq Tilt\ angle < O_{max}{}^{k} \quad \text{Equation 6-19}$$

$$0° \leq Pan\ angle \leq 360° \quad \text{Equation 6-20}$$

where: L is the maximum value of the length of the placement surface, H_{min} and H_{max} are the minimum and maximum values of the height of the camera on the wall, respectively, W is the maximum value of the width of the placement surface.

Similarly, this chapter uses the previous adopted GA (i.e. PICEA) in order to address the placement problem. Using the crossover and mutation operators (Senaratna, 2005), the next generations are created and sent one by one to the visibility simulation process to calculate the objective function. In case the crossover operator is not applied, one of the parents will be copied and may be prepared to a mutation process. This process ends by finding the near-optimum solution, which can ensure the best *CP* coverage.

6.3 Implementation and case study

An integrated system was developed in order to validate the proposed method. A case study was conducted on parts of the ground floor of the Engineering and Visual Art building in Concordia University using the same BIM model of Chapter 4. The case study aims to optimize the placement of one PTZ camera in a BIM model. The floor has some physical constraints, such as columns and stairs that can affect the PTZ camera coverage. The BIM model was imported into the Unity3D (Unity, 2015) simulation module using FBX format for facilitating the computational processes, which were developed using C# language. In Unity3D, the floor of the model was divided into a number of important areas (areas a_1 to a_7 as shown in Figure 6-6). The important areas were covered by equal 3D cubic cells in a way that covers the upper limit of the average human height. These cells were assigned different importance values based on their locations. Importance values of 1, 3 and 6 represent the lowest, medium and highest importance, respectively. For example a_6 is assigned the importance value 6 because it represents the entrance from the basement to the ground floor. The optimum PTZ camera placement is where the maximum *CP* coverage can be guaranteed using a number of phases. The dimensions of the monitored area are 14 m on the *X* axis, 49 m on the *Z* axis and 5 m on the *Y* axis. A grid system is attached to the ceiling surface with a step of 25 cm in order to define the camera placement. The grid forms 56×196 possible PTZ camera locations. The implementation integrates the GA optimization module using Matlab (Matlab, 2015) and the simulation module and has the following steps:

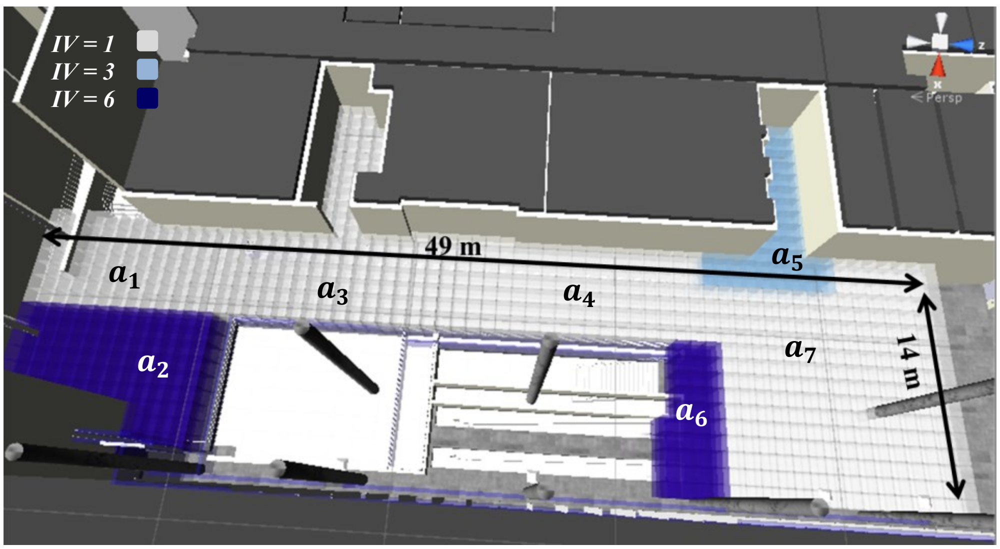


Figure 6-6 Monitored areas with importance values and the placement surface

(1) Obtaining the valid coordinates of the PTZ camera placement from BIM model. This includes specifying the decision variables (i.e. *X*, *Y*, *Z*, pan and tilt angles) that will be used in the GA module in order to define the search space, as explained in Section 5.2.1.2. Autodesk Revit is used to specify the valid locations of the PTZ camera placement. The valid locations are sent to Matlab to start the optimization process.

(2) Determining the population size and the number of generations, the ranges of phase and travel time and the cycle path time using the GA interface. The FOV angle of the PTZ camera is set to 90°. The decision variables that can affect the optimization are as the following: *X, Y, Z* for the camera location; pan and tilt angles for the orientation; number of phases; travel times; and phase times.

(3) Initializing the population of the first generation. The potential solutions are saved into a text file. The text file is exported to the simulation module to calculate the objective function. A sample of the text file is given in Appendix I.

(4) The simulation module receives the text file and conducts the camera placement for each solution in the text file.

(5) The objective function is calculated based on the visibility analysis. The code of calculating *CP* coverage in the game engine is given in Appendix E.

(6) The *CP* coverage results are saved into a text file and returned to the GA module.

(7) The GA assesses the results of the *CP* coverage and conducts the crossover and mutation operations to generate new populations. This process continues until the maximum number of generations is reached. The final results include the sets of PTZ camera placements with the best fitness values.

The parameters of the GA system are selected as the following: The maximum number of generations = 65, the population size = 250, the crossover rate = 90% and mutation rate = 5% and the cycle path time = 90 sec. The acceptable overlapping percentage is set to 50%. The pan movement is selected to be the dominant movement of the FOV, and results in 4 phases. The tilt angle is set between 0° to 90° and $({T_{p_k}}^{max})_{min} \geq 22.5$ sec. The range of T_{p_k} is set from 5 to 25 sec. Although $T_{t_k}$is affected by the speed of the camera, for simplicity it is fixed to 3 sec. The orientation in the pan direction is restricted between 0° and 45°, 90° and 135°, 180° and 225° and 270° and 315° for phases 1, 2, 3 and 4, respectively.

Figure 6-7 shows the best fitness and the mean fitness curves for each generation. The best fitness curve shows an increase in the *CP* coverage from the first generation to generation 65. The best fitness value is 43.19%, which existed from generation 63 to generation 65. The best fitness curve shows a number of fluctuations throughout different generations. For example, this curve shows a fluctuation in the *CP* coverage ranges between 21% to 26% from generation 31 to generation 36. The curve gained *CP* coverage of 34% in generation 52, and kept increasing until it reaches the maximum *CP* coverage in generation 63. Table 6-1 shows the four optimum solutions, depicted in Figure 6-7, with their *CP* coverage results. The case study proved the capability of the proposed method to provide near-optimum results with an acceptable overlapping percentage and a logical sequence of the phases. It is important to note that the best results were produced by placing the camera near the edge of the placement surface and by assigning more phase time to the phases that cover cells with high importance values. Figure 6-8 shows the 3D simulation of Solution 1, which represents the near-optimum solution of this case study. This solution provides 4 phases with the camera location at *X* = 100 cm, *Y* = 500 cm, *Z* = 3,900 cm. The time of phases is 25, 25, 9 and 19 sec. for phases 1 to 4, respectively. This solution offered the *CP* coverage of 43.19%. Figures 6-9, 6-10 and 6-11 show the 3D simulation of solutions 2, 3 and 4, respectively.

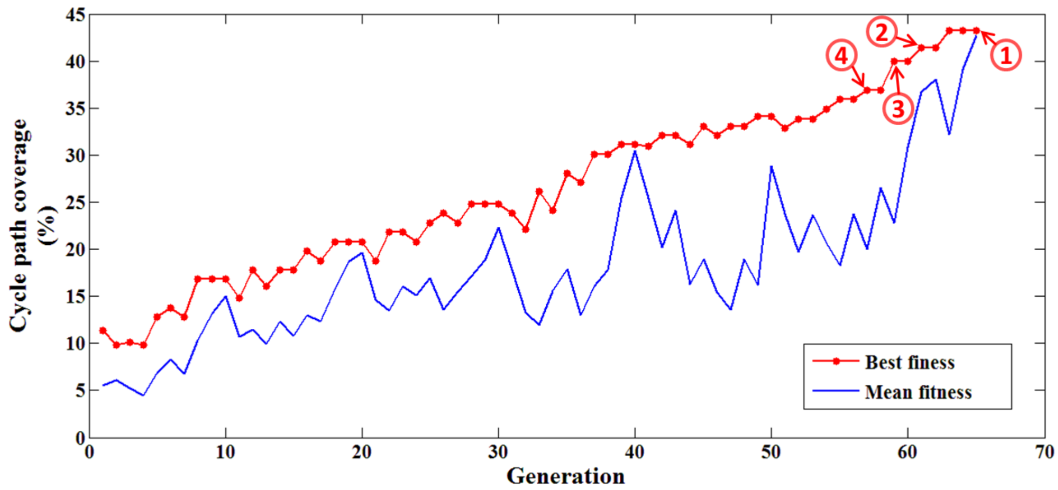
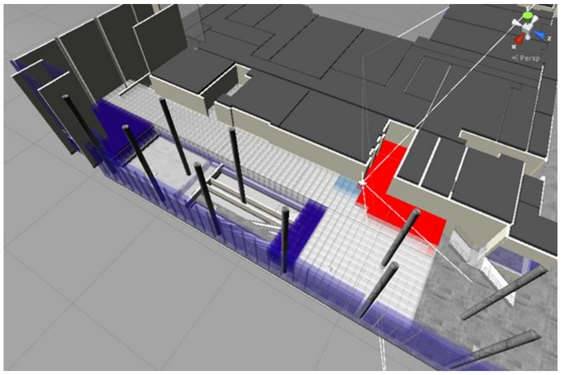


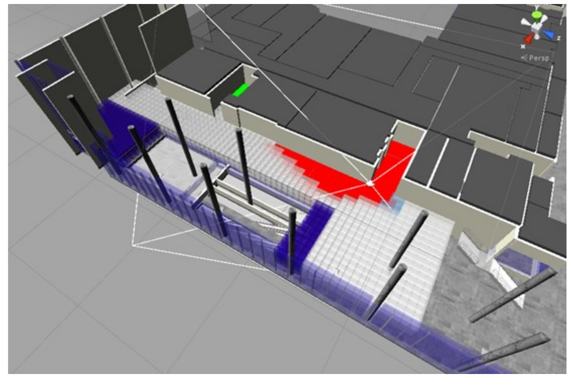
Figure 6-7 Fitness values throughout generations

Table 6-1 Best four solutions of the optimization process

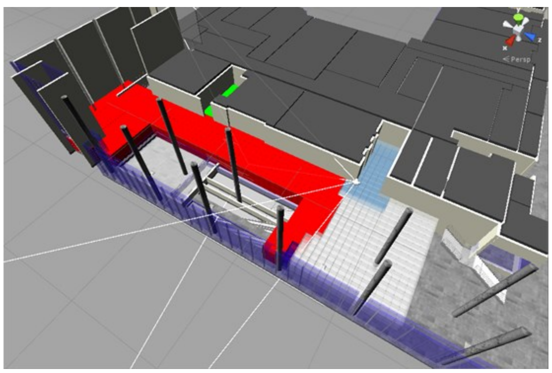
Solution #	*Camera parameters*							*CP coverage (%)*
	X (cm)	Y (cm)	Z (cm)	Phase time (*sec.*)	Pan (°)	Tilt (°)	Phase coverage-time (*cell×sec.*)	
1	100	500	3,900	25	26	69	9,852	43.19
				25	102	71	9,113	
				9	180	56	105,574	
				19	277	68	86,523	
2	1,350	500	4,050	24	44	73	8,571	41.43
				15	105	47	81,596	
				14	183	56	106,575	
				25	271	88	5,695	
3	100	500	4,000	25	38	79	8,468	39.97
				24	108	66	7,464	
				7	189	69	112,560	
				22	280	47	66,835	
4	1,350	500	3,900	25	31	50	8,845	36.94
				13	130	56	62,776	
				15	183	62	102,473	
				25	274	90	6,425	



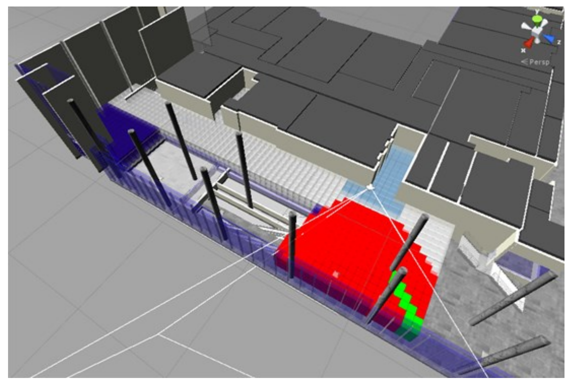
(a) Phase 1



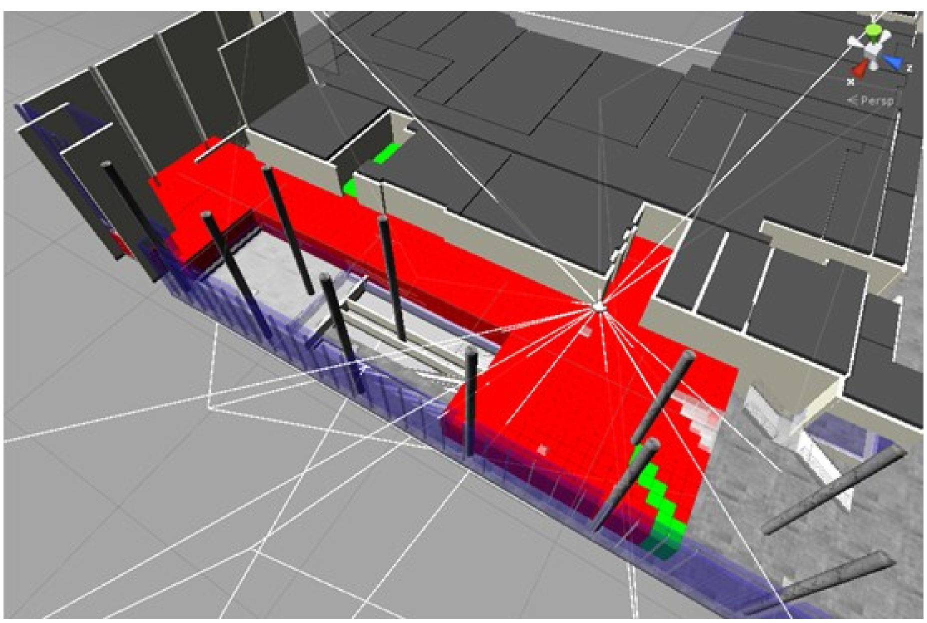
(b) Phase 2



(c) Phase 3



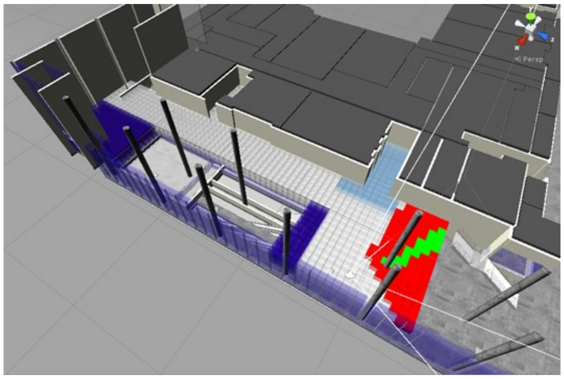
(d) Phase 4



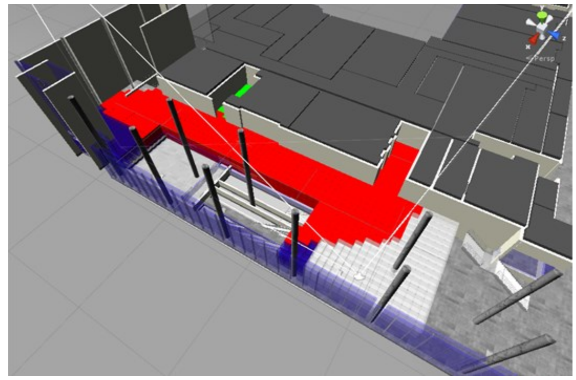
(e) All phases

Visible cells   Invisible cells

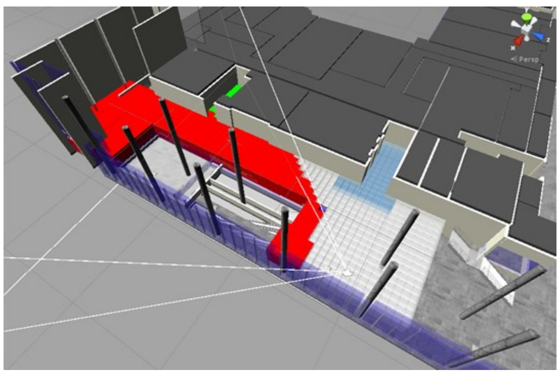
Figure 6-8 Visibility simulation of solution 1



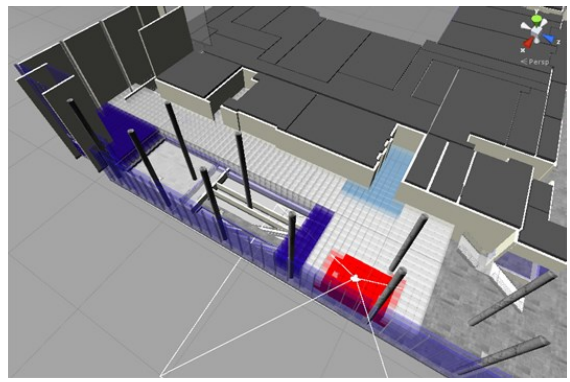
(a) Phase 1



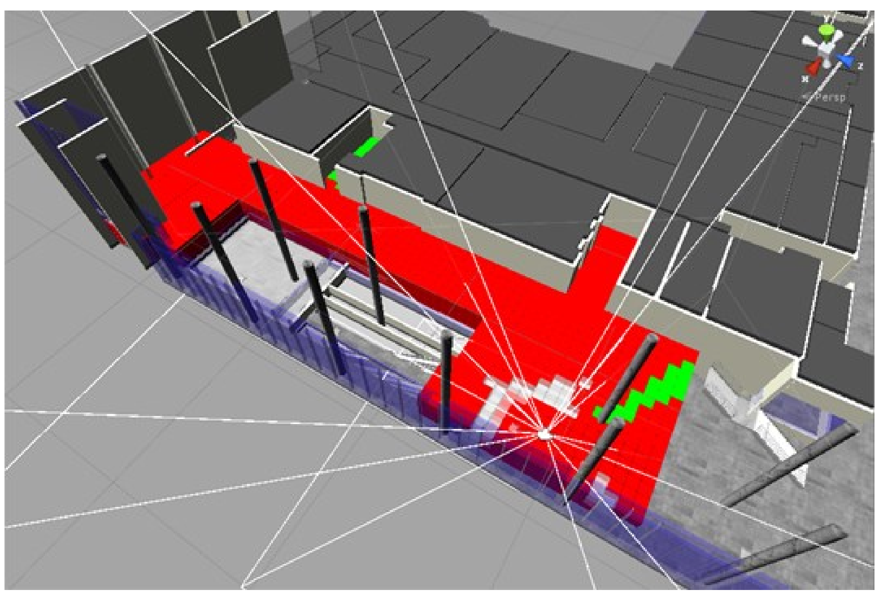
(b) Phase 2



(c) Phase 3



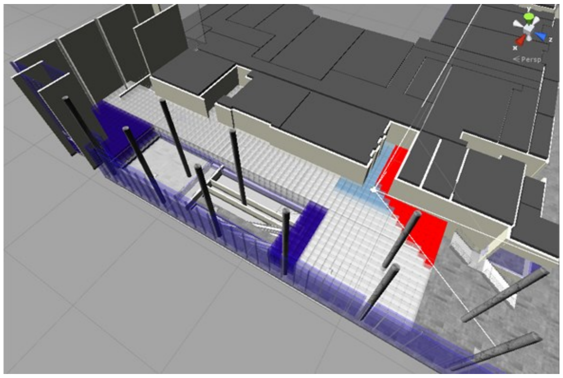
(d) Phase 4



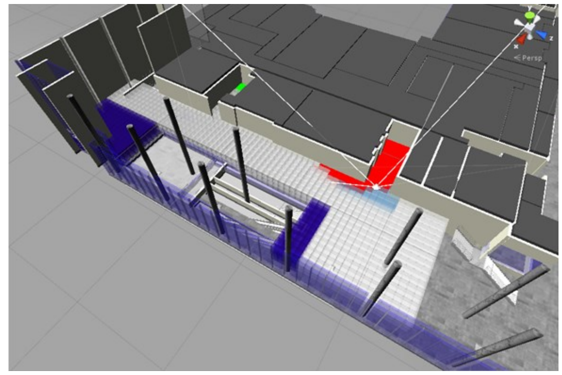
(e) All phases

Visible cells Invisible cells

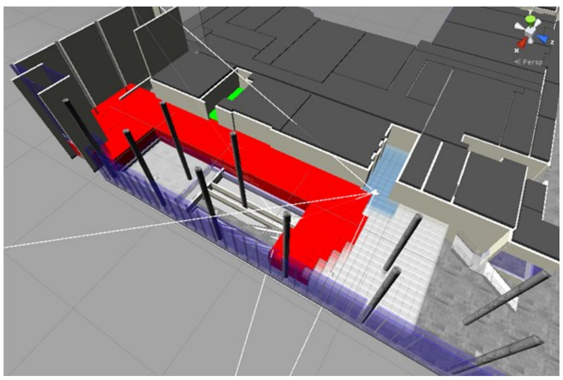
Figure 6-9 Visibility simulation of solution 2



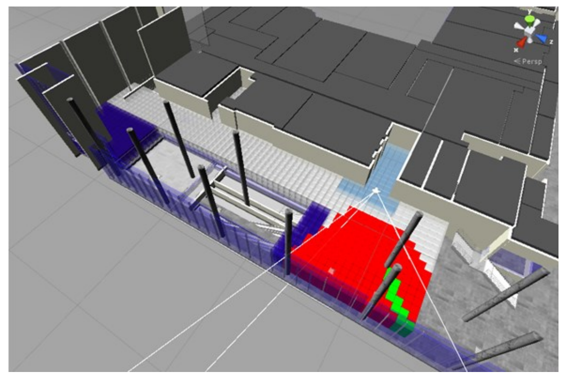
(a) Phase 1



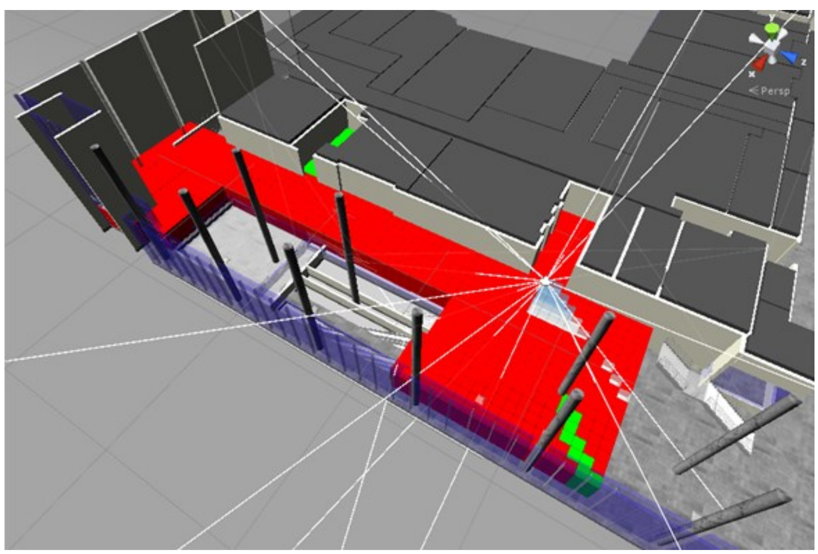
(b) Phase 2



(c) Phase 3



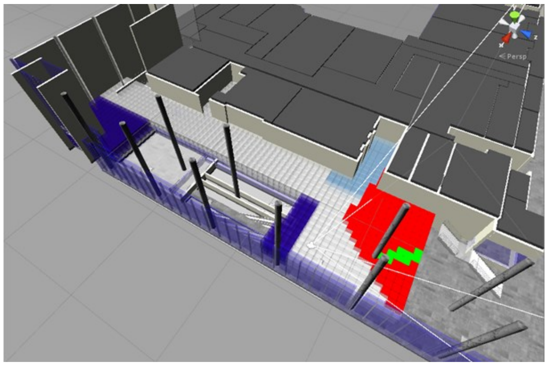
(d) Phase 4



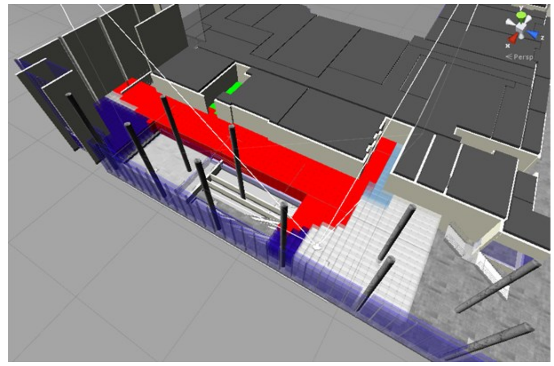
(e) All phases

Visible cells   Invisible cells

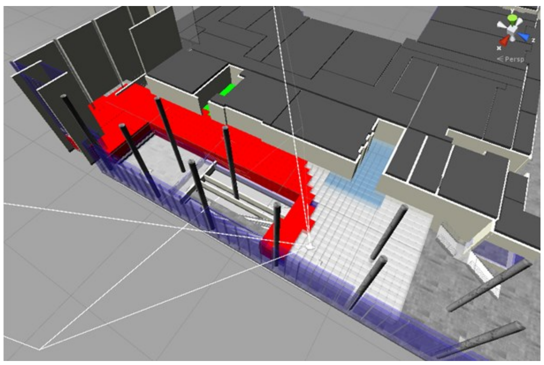
Figure 6-10 Visibility simulation of solution 3



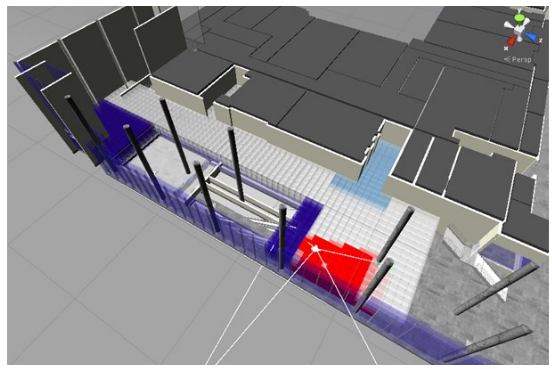
(a) Phase 1



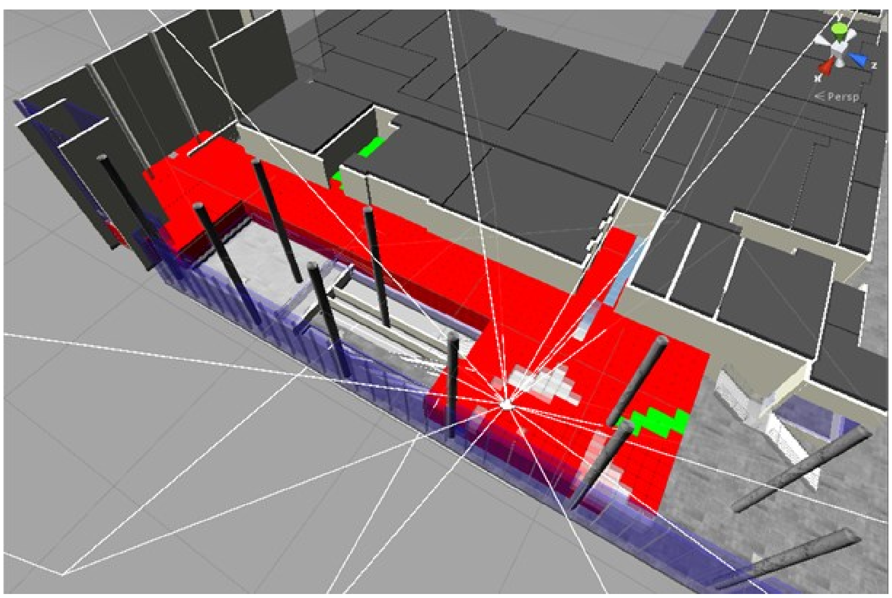
(b) Phase 2



(c) Phase 3



(d) Phase 4



(e) All phases

Visible cells   Invisible cells

Figure 6-11 Visibility simulation of solution 4

6.4 Summary and conclusions

This chapter presented a simulation-based optimization method for the placement of a PTZ camera in a building using BIM. The near-optimum placement aims to find the best phases that can maximize the *CP* coverage. The *CP* coverage is the average coverage of the PTZ camera during its *CP* time. BIM tool is used to define the valid locations of the camera placement. The proposed method integrates a 3D simulation system for calculating the *CP* coverage of the PTZ camera and GA optimization. The method includes the following steps: (1) specifying the monitored area in BIM tool; (2) applying GA optimization; and (3) the visibility simulation process. The following steps were conducted to calculate the *CP* coverage: (1) placing the PTZ camera; (2) detecting the importance values of cells in each phase; (3) calculating the coverage-time of each phase; and (4) calculating the *CP* coverage. Also, the study considered the effect of overlapping between phases. The validity of the proposed method was proven using a case study. The maximum *CP* coverage achieved was 43.19%. The main contribution of the chapter is to propose a method that can find the near-optimum placement and movement plan of a PTZ camera inside buildings taking into consideration the existence of different geometrical constraints on the placement surface. The proposed method can help individuals who are responsible of installing PTZ cameras to efficiently determine the optimum camera placement and phases that can ensure best *CP* coverage.

CHAPTER 7: CONCLUSIONS AND FUTURE WORK

7.1 Summary and conclusions of the research work

The main purpose of this research is to establish a novel method that can optimize the placement of different types of cameras inside building. To achieve this purpose, the research conducted the following tasks: (1) developing a method to calculate the camera coverage inside buildings; (2) developing a method to optimize the placement of fixed cameras inside buildings; and (3) extending the previous methods to optimize the placement of a PTZ camera inside buildings. Several methods have been developed to realize these tasks. These methods have been implemented using simulation and optimization tools. A number of case studies were conducted in order to demonstrate the applicability of the proposed methods.

Chapter 4 presented a method of calculating the camera coverage inside building. This method included: (1) defining the camera placement; (2) validating the camera scenes; (3) generating cells and assigning importance values (4) visibility analysis; and (5) calculating the camera coverage. The chapter included a sensitivity analysis of evaluating the suitable cell size which used to cover the monitored area. A case study was conducted to validate the method of calculating the camera coverage. This included four scenarios that focused on evaluating the effect of the FOV and β angles on the camera coverage. The results proved the ability of the FOV and β angle to increase the coverage to 89.40%. The experiment proves the necessity of considering the height of the camera during the process of calculating the coverage in order to detect invisible spots behind obstacles.

Also, dividing the monitored area into a number of important areas with different (*IV*) values facilitated the decision of selecting the proper FOV and β angles for the camera.

Chapter 5 presented a new method of finding the near-optimum number, type and placement for surveillance cameras inside buildings. The optimal placement of cameras ensures maximum camera coverage and minimum cost. Furthermore, BIM tool is used to specify the valid locations of the camera on the placement surface. The proposed method integrated the simulation of the camera coverage and GA optimization. The method included the following steps: (1) specifying the valid coordinates, (2) generating solutions, (3) coverage simulation, (4) evaluating the results, and (5) visualizing the optimum results. The feasibility of the method was demonstrated using two case studies: (1) optimizing the placement of a single camera; and (2) optimizing the placement of multiple camera types. The results of the case studies showed that the maximum coverage can be achieved by placing the cameras near the edge of the placement surface with the appropriate pan and tilt angles. Also, the results showed the ability to increase the coverage by adding more cameras with different FOV angles. The maximum coverage was 92% by adding four cameras each with FOV = 100° wich corresponds to the maximum cost.

Chapter 6 presented a simulation-based optimization method for the placement of PTZ camera in a building using BIM. The optimal placement aims to find the best number of phases that can maximize the *CP* coverage. The proposed method integrated a 3D simulation system for calculating the *CP* coverage of the PTZ camera and GA optimization. The method included the following steps: (1) preparing the area that will be monitored by the PTZ camera, (2) specify the search space using BIM tool, (3) implementing the GA optimization process, and (5) calculating the cycle path coverage. Also, the method considered the importance to respect the logical sequence and overlapping between phases. The feasibility of the methods was demonstrated using a case study. The results of the case study showed the near-optimum phases that can offer the maximum cycle path coverage. The maximum cycle path coverage was 43.19%.

The proposed methods can help individuals who are responsible of the camera installation to efficiently determine the near-optimum types, number and placement of cameras needed to monitor spaces in buildings.

7.2 Research contributions

Expected contributions of this research can be listed as follows:

(1) Developing a method that can calculate the camera coverage inside buildings taking into account the existence of different obstructing elements that can affect the camera coverage.

(2) Developing a method that can find near-optimum placement of single surveillance camera inside building taking into consideration the existence of different geometrical constraints on the placement surface.

(3) Developing a method that can find near-optimum for the number, type and placement of multiple fixed surveillance cameras inside building.

(4) Developing a method that can find the near-optimum placement of a PTZ camera inside buildings. The method considered the possibility of overlapping between different phases sharing the same cycle path.

7.3 Limitations and recommendations for future work

This research has a number of limitations as the following:

(1) Data from real cases (e.g. FOV, direction of the cameras and the blind spots) that could be useful to compare with the results of this research were not available due to security concerns. Future work may include accurate data of real projects after obtaining the suitable permits.

(2) The research conducted different case studies assuming rectangular shapes for the placement surface of the cameras. Future work may consider more complex geometry.

(3) The case studies used the Preference-Driven Co-evolutionary Algorithm for the optimization process. Other metaheuristic methods can be applicable for addressing the camera placement problem. Future work will include an evaluation for a number of metaheuristic methods (e.g. particle swarm optimization) to optimize the camera placement inside buildings.

(4) Further effort is needed regarding the constraints of the placement surface and the way to determine the suitable size of their buffers in BIM tool. To address this problem, a survey for collecting data from experts can be conducted.

(5) The current research optimizes the placement of multiple fixed cameras and a single PTZ camera. Future work will include the optimization of a network of multiple PTZ cameras and other surveillance systems such as the omnidirectional cameras to achieve the best security coverage.

(6) The scope of this research is limited to optimizing the cameras' placement inside buildings. The optimization problem can be developed to be used for cases where the cameras are placed outside the buildings or in construction sites.

(7) The research considered the FOV as the basic feature that can affect the type of the camera. The research can be extended to consider more features of the camera in the optimization problem such as the camera resolution, which is affected by the size and type of the camera sensor.

(8) The selection of importance values is based on experience. Path planning algorithms can be integrated with the proposed methods to accurately determine the level and location of important areas on the monitored area.

(9) The current research is limited to surveillance systems. The proposed methods can be used to optimize the placement for other types of devices inside smart buildings such as smoke and motion detectors.

REFERENCES

Abraham, A., Jain, L., and Goldberg, R. (eds.,) (2005). Evolutionary Multiobjective Optimization: Theoretical Advances and Applications. 1st Ed., Springer-Verlag London Limited, London.

Adeli, h., and Karim, A., (1997). "Scheduling/Cost Optimization and Neural Dynamics Model for Construction Projects." Journal of Construction Engineering and Management, Vol. 123(4): 450-458.

Albahri, A., and Hammad, A. (2015). Evaluating the Coverage of Surveillance Cameras Using BIM and Game Engine. In 2st International Conference on Civil and Building Engineering Informatics (ICCBEI 2015).

Amriki, K. A., and Atrey, P. K. (2012). Bus surveillance: how many and where cameras should be placed. Multimedia Tools and Applications, 1-35.

Anukul, S. (2015). PTZ Camera Deployments. The Right and the Wrong?. Extracted on 5 Jan 2017 from: < http://olivechats.com/t/ptz-camera-deployments-the-right-and-the-wrong/413>.

Asen, Y. (2012), Building Information Modeling Based Integration and Visualization for Facilities Management. MASc Diss. Concordia University.

Autodesk, (2014). Easily Find, Preview and Download High Quality BIM Models and DWG Files. Extracted on 16 Sep 2014. From:< http://seek.autodesk.com/>

Axis Communications (2013). Axis Camera Families Now Available For Autodesk Seek While New Axis Camera View Add-In Provides 3D Camera Perspectives For Autodesk Revit Users. Innovative Retail Technologies. Extracted on 11 Nov 2014

<http://www.innovativeretailtechnologies.com/doc/axis-camera-available-autodesk-whilenew-perspectives-revit-users-0001>

Axis Communication (2015). Network Cameras. Extracted on 22 Dec 2015 from: < http://www.axis.com/global/en/products/network-cameras >

Azhar, S. Nadeem, A. Mok, J. and Leung, B. (2008), Building Information Modeling (BIM): A new paradigm for visual interactive modeling and simulation for construction projects. Proceedings of the First International Conference on Construction in Developing Countries.

Bandyopadhyay, S. and Pal, S.K., (2007). Classification and Learning Using Genetic Algorithms: Applications in Bioinformatics and Web Intelligence. Springer Science and Business Media.

Barlish, K., and Sullivan, K. (2012). How to Measure the Benefits of BIM. A Case Study Approach. Automation in construction, Vol. 24: 149-159.

Becker, E., Guerra-Filho, G., and Makedon, F. (2009). Automatic Sensor Placement in a 3D Volume. In Proceedings of the 2nd International Conference on Pervasive Technologies Related to Assistive Environments (36). ACM.

Bernard, R. (2014). Convergence Q and A: Convergence in Design: Virtual Construction. Security InfoWatch. Extracted on 25 Feb 2016 <http://www.securityinfowatch.com/article/11476902/bim-modeling-is-not-evolution-of-security-systems-planning>).

Bigdeli, A., Lovell, B. C., Sanderson, C., Shan, T., and Chen, S. (2007). Vision processing in intelligent CCTV for mass transport security. In IEEE Workshop on Signal Processing Applications for Public Security and Forensics. SAFE'07. (1-4). IET.

BIMhub. (2013). Challenges in BIM Implementation. Extracted at 14 Sep 2014. From: <http://www.bimhub.com/blog/challenges-bim-implementation/>

Bodor, R., Drenner, A., Schrater, P., and Papanikolopoulos, N. (2007). Optimal Camera Placement for Automated Surveillance Tasks. Journal of Intelligent and Robotic Systems, Vol. 50(3): 257-295.

Bohn, J. S., and Teizer, J. (2009). Benefits and Barriers of Construction Project Monitoring Using High-Resolution Automated Cameras. Journal of Construction Engineering and Management, 136(6), 632-640.

BOMA Calgary Public Safety Committee. (2007). Laptop theft in commercial buildings: 2007 survey. Available: <http://www.boma.ca/Surveys/ 2007LaptopTheftSurvey.pdf>.

BOSH. (2015). Bosch to provide BIM files to facilitate construction. Improving efficiency for contractors and building designers. Extracted on 10 Jan 2016 <http://us.boschsecurity.com/en/05_news_and_extras_2/05_04_press_releases_2/2015_1/bosch_to_provide_bim_files_to_facilitate_construction/bosch_to_provide_bim_files_to_facilitate_construction>.

Bremer, D. (2011). BIM and Security System Design. Modeling Capabilities Demonstrate Security Integration. Contractor Power and Integrated Building Systems. Extracted on 10 Nov 2015 <http://www.ecmag.com/section/systems/bim-and-security-system-design>.

California Department of Transportation. (1994). Guidelines for Camera Placement. IVHSROS Systems in the United States," by IVHS America, Design Guideline 94-01, California.

Cassidy, K., (2009). "Camera Movement Techniques - Tilt, Pan, Zoom, Pedestal, Dolly and Truck." Videomaker. Extracted at 3 Oct 2013. From: <http://www.videomaker.com/article/14221-camera-movement-techniques-tiltpan-zoom-pedestal-dolly-and-truck>

Chae, S., and Kano, N. (2007). Application of Location Information by Stereo Camera Images to Project Progress Monitoring. In Proceedings of the 24th International Symposium on Automation and Robotics in Construction, Kochi, Kerala, India (pp. 19-21).

Check video (2014). Camera Placement Guide. Check video. USA

Chelliger, D., (2008). From the Ground Up: Security for Tall Buildings. CRISB Report. Connecting Research in Security to Practice. In ASIS International. Alexandria, VA, USA.

Chen, H. T., Wu, S. W., and Hsieh, S. H. (2013). Visualization of CCTV Coverage in Public Building Space Using BIM Technology. Visualization in Engineering, Vol. 1(1):1-17.

Chen, X. and Davis, J. (2000). Camera Placement Considering Occlusion for Robust Motion Capture. Computer Graphics Laboratory, Stanford University, Tech. Rep, 2(2.2):2.

Cole, R. and Sharir, M. (1989). Visibility Problems for Polyhedral Terrains. Journal of Symbolic Computation, 7(1):11-30.

Cohen, N., Gattuso, J., and MacLennan-Brown, K. (2009). CCTV Operational Requirements Manual. Home Office Scientific Development Branch, UK.

Cohon, J. L., (1978). Multiobjective Programming and Planning. Academic Press, New York.

Cowan, C. K., and Kovesi, P. D. (1988). Automatic Sensor Placement from Vision Task Requirements. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 10(3): 407-416.

CRC Construction Innovation. (2007). Adopting BIM for Facilities Management: Solutions for Managing the Sydney Opera House, Cooperative Research Center for Construction Innovation, Brisbane, Australia.

Deb, K., (2001). Multi-objective Optimization Using Evolutionary Algorithms. 1st Ed., Wiley, Chichester, U.K.

De Freitas, A. (2012). Matlab Central: Open Genetic Algorithm Toolbox. Extracted at 15 Oct 2015 from: <http://www.mathworks.com/matlabcentral/fileexchange/37998-open-genetic-algorithm-toolbox>

Dirksen, J. (2013). Working with the Basic Components that Make Up a Three.js Scene. Extracted on 15 Nov 2016 from: < https://www.packtpub.com/books/content/working-basic-components-make-threejs-scene>.

Diwekar, U., (2008). Introduction to Applied Optimization. 2nd Ed., Springer Science+Business Media, New York.

Dornaika, F. and Elder, J.H., (2012). Image Registration for Foveated Panoramic Sensing. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 8(2), p.17.

Douglas, A. Kerr, P. (2006). Field of View in Photography. Extracted on 18 Nov 2016 from: <http://dougkerr.net/Pumpkin/articles/Field_of_View.pdf>.

Eastman, C., Teicholz, P., Sacks, R., and Liston, K. (2011). BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers, and Contractors. Second edition, Wiley, Hoboken, NJ.

Electrophysics Introduces, (2006). How Night Vision Works. Extracted on 17 Nov 2016 from: <http://www.nightvisioncameras.com/dl-files/newsletters/nv_2006_hnvw.pdf>.

Erdem, U. M., and Sclaroff, S. (2006). Automated Camera Layout to Satisfy Task-Specific and Floor Plan-Specific Coverage Requirements. Computer Vision and Image Understanding, Vol. 103(3): 156-169.

Fehr, D., Fiore, L., and Papanikolopoulos, N. (2009). Issues and Solutions in Surveillance Camera Placement. In Intelligent Robots and Systems. IEEE/RSJ International Conference (pp. 3780-3785).

Feng, C.-W., Liu, L., and Burns, S. A., (2000). "Stochastic Construction Time-Cost Trade-Off Analysis." Journal of Computing in Civil Engineering, Vol. 14(2): 117-126.

Gigliotti, R., and Jason, R., (1984). Approaches to Physical Constraints. From: Fennelly, L. (1992). Effective Physical Security. Butterworth-Heinemann.Security and Crime Prevention. Butterworth-Heinemann.

Goh, C.-K., and Tan, K. C., (2009). Evolutionary Multi-objective Optimization in Uncertain Environments. Springer-Verlag Berlin Heidelberg, Berlin.

Gonzalez-Barbosa, J. J., García-Ramírez, T., Salas, J., Hurtado-Ramos, J. B., and Rico-Jiménez, J. D. J. (2009). Optimal Camera Placement for Total Coverage. In Proceedings of the IEEE international conference on Robotics and Automation (pp. 3672-3676). IEEE Press.

Green, M. (1999). The Appropriate and Effective Use of Security Technologies in U.S. Schools. A Guide for Schools and Law Enforcement Agencies. U.S. Department of Justice Office of Justice Programs. Washington.

Gu, N., and London, K. (2010). Understanding and Facilitating BIM Adoption in the AEC Industry. Automation in construction, Vol. 19(8): 988-999.

Hergunsel, M. F. (2011), Benefits of Building Information Modeling for Construction Managers and BIM Based Scheduling. Master Thesis, Worcester Polytechnic Institute.

Hi-tech hi-rise. (2002). How the Flats of the Future will Fight the Crimes of Today. Design Council Newsletter, London.

Honovich, J., (2009). Should you use PTZ cameras? Security Info-Watch. Extracted on 20 April 2016: <http://www.securityinfowatch.com/article/10492590/should-you-use-ptz-cameras>.

Janoos, F., Machiraju, R., Parent, R., Davis, J.W. and Murray, A., (2007). Sensor Configuration for Coverage Optimization for Surveillance Applications. In Electronic Imaging 2007, pp. 649105-649105. International Society for Optics and Photonics.

Jimenez, H. A. T., (2007). Multi-objective Optimization Algorithms Considering Objective Preferences and Solution Clusters, PhD Thesis, The State University of New Jersey.

Kelly, H. (2013), After Boston: The Pros and Cons of Surveillance Cameras, CNN Tech. Extracted at 25 Sep 2013. From: <http://www.cnn.com/2013/04/26/tech/innovation/security-cameras-boston-bombings/>.

Kim, K., Murray, A.T. and Xiao, N., (2008). A Multiobjective Evolutionary Algorithm for Surveillance Sensor Placement. Environment and Planning B: Planning and Design, 35(5): 935-948.

Krahnstoever, N., Yu, T., Lim, S. N., Patwardhan, K., and Tu, P. (2008). Collaborative Real-Time Control of Active Cameras in Large Scale Surveillance Systems. In Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications-M2SFA2.

Kruegle, H. (1992). Closed Circuit Television Security. From: Fennelly, L. (1992). Effective physical security. Butterworth-Heinemann.Security and Crime Prevention. Butterworth-Heinemann.

Kumar, S. (2012). What are Pan, Tilt, and Zoom Cameras?. CCTV Camera Dealer and System Integraters, Kerala. Extracted on 24 Feb 2015 from:

< http://securitysystemsinkerala.blogspot.ca/2012/11/what-are-pan-tilt-and-zoom-cameras.html>.

Lalonde, M., Foucher, S., Gagnon, L., Pronovost, E., Derenne, M. and Janelle, A., (2007). A System to Automatically Track Humans and Vehicles with a PTZ Camera. In Defense and Security Symposium (pp. 657502-657502). International Society for Optics and Photonics.

Lee, L. K., Zachariah, M., and Everett, P. (1995). CCTV Camera Site Selection: A Field Experience. In Vehicle Navigation and Information Systems Conference, Proceedings In conjunction with the Pacific Rim TransTech Conference, IEEE, the 6th International VNIS.'A Ride into the Future, (pp. 21-27).

Lee, J.Y., Seok, J.H. and Lee, J.J., (2012). Multiobjective Optimization Approach for Sensor Arrangement in a Complex Indoor Environment. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(2): 174-186.

Liu, L., Burns, S., and Feng, C., (1995). "Construction Time-Cost Trade-Off Analysis Using LP/IP Hybrid Method." Journal of Construction Engineering and Management, Vol. 121(4): 446–454.

Mansurov, N. (2015). Equivalent Focal Length and Field of View. Extracted on 6 Jan 2017 from: <https://photographylife.com/equivalent-focal-length-and-field-of-view>.

Maver, J., and Bajcsy, R. (1993). Occlusions as a Guide for Planning the Next View. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 15(5), 417-433.

Mawlana, M., (2015), Stochastic Simulation-based Optimization of Precast Box Girder Bridge Construction Projects. PhD Diss. Concordia University.

Matlab (2015). Mathworks. Extracted on: 20 Aug 2015 from: < http://www.mathworks.com/products/matlab/>

Moog Videolarm (2012). IP Camera Protection: Get the Heat Out. A Discussion on Heat and Security Cameras. USA.

Morsly, Y., Aouf, N., Djouadi, M.S. and Richardson, M., 2012. Particle Swarm Optimization Inspired Probability Algorithm for Optimal Camera Network Placement. IEEE Sensors Journal, 12(5), 1402-1412.

Motamedi, A., and Hammad, A., (2009). Lifecycle Management of Facilities Components Using Radio Frequency Identification and Building Information Model, Next Generation Construction IT, Journal of IT in Construction. Vol. 14: 238-262.

Murray, A. T., Kim, K., Davis, J. W., Machiraju, R., and Parent, R. (2007). Coverage Optimization to Support Security Monitoring. Computers, Environment and Urban Systems, Vol. 31(2):133-147.

Nakayama, H., Yun, Y., and Yoon, M. (2009). Sequential Approximate Multiobjective Optimization Using Computational Intelligence. Springer-Verlag Berlin Heidelberg, Berlin.

Nam, Y., and Hong, S. (2014). Optimal Placement of Multiple Visual Sensors Considering Space Coverage and Cost Constraints. Multimedia Tools and Applications, Vol. *73*(1): 129-150.

National Institute of Low Enforcement and Criminal Justice. (1974). Standards for the Physical Security of Door Assemblies and Components. U.S. Department of Justice, Washington.

Ogino, S., Tomioka, Y., and Kitazawa, H. (2014). Patrol Course Planning and Battery Station Placement for Mobile Surveillance Cameras. In Multimedia and Expo (ICME), 2014 IEEE International Conference on (pp. 1-6). IEEE.

O'Block, R. L. (1981). Environmental Design. From: Fennelly, L. (1992). Effective Physical Security. Butterworth-Heinemann.Security and Crime Prevention. Butterworth-Heinemann.

OPTICOM (2013). Video Installation Tips: Solutions for High Vibration Applications. USA.

O'Rourke, J. (1987). Art Gallery Theorems and Algorithms (Vol. 57). Oxford: Oxford University Press.

Pelco. (2014). For Any Application, Any Lighting Condition, Any Environment. Extracted at 26 Feb 2014. From: < https://www.pelco.com/ar/video-surveillance-camera-security-systems>

Purshouse, R.C., Jalba, C. and Fleming, P.J. (2011). Preference-Driven Co-evolutionary Algorithms Show Promise for Many-Objective Optimisation. In Evolutionary Multi-criterion Optimization (EMO), pp. 136-150.

Qureshi, F.Z. and Terzopoulos, D., (2007). Surveillance in Virtual Reality: System Design and Multi-Camera Control. In 2007 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.

Qureshi, F. Z., and Terzopoulos, D. (2009). Planning Ahead for PTZ Camera Assignment and Handoff. In Distributed Smart Cameras, ICDSC 2009, Third ACM/IEEE International Conference, 1-8.

Reddy, M., and Kumar, D., (2007). Multiobjective Differential Evolution with Application to Reservoir System Optimization. Journal of Computing in Civil Engineering, Vol. 21(2): 136-146.

Reed, M. K., and Allen, P. K. (2000). Constraint-Based Sensor Planning for Scene Modeling. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 22(12), 1460-1467

Reid, R. N. (2005). Facility Manager's Guide to Security: Protecting Your Assets. The Fairmont Press, Inc.

Senaratna, N.I., 2005. Genetic Algorithms: The Crossover-Mutation Debate. Degree of Bachelor of Computer Science of the University of Colombo.

Senouci, A., and El-Rayes, K., (2009). Time-Profit Trade-Off Analysis for Construction Projects. Journal of Construction Engineering and Management, Vol. 135(8): 718-725.

Sinha, S.N. and Pollefeys, M., (2006). Pan–Tilt–Zoom Camera Calibration and High-Resolution Mosaic Generation. Computer Vision and Image Understanding, 103(3), pp.170-183.

Sjogren, J. and Kvarsvik, K. (2007). BuildingSMART Standars. Information and Demonstration. EUREKABUILD Workshop.

Spanier, G. B., and Fishel, C. (1973). The Housing Project and Familial Functions: Consequences for Low-Income, Urban Families. Family Coordinator, (pp. 235-240).

Spletzer, J. R., and Taylor, C. J. (2001). A Framework for Sensor Planning and Control with Applications to Vision Guided Multi-Robot Systems. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on (Vol. 1, pp. I-378). IEEE.

Starzyk, W., and Qureshi, F. Z. (2011). Multi-Tasking Smart Cameras for Intelligent Video Surveillance Systems. In the 8th IEEE International Conference Advanced Video and Signal-Based Surveillance (AVSS): (pp. 154-159).

Stellitano, C. (2005). Janitors Against Terrorism. Access Control and Security Systems, Vol. 48(2): 18.

Tarabanis, K., Tsai, R. Y., and Kaul, A. (1996). Computing Occlusion-Free Viewpoints. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 18(3), 279-292.

Tomioka, Y., Takara, A., and Kitazawa, H. (2010). Travelling Route of Mobile Surveillance Camera. In Circuits and Systems (APCCAS), 2010 IEEE Asia Pacific Conference on (pp. 1043-1046). IEEE.

Tomioka, Y., Takara, A., and Kitazawa, H. (2011). Enhanced Patrol Course Planning Method for Multiple Mobile Surveillance Cameras. In Distributed Smart Cameras (ICDSC), 2011 Fifth ACM/IEEE International Conference on (pp. 1-6). IEEE.

Toregas, C., Swain, R., ReVelle, C., and Bergman, L. (1971). The Location of Emergency Service Facilities. Operations Research, Vol. 19(6): 1363-1373.

Unity, (2015). Download and Start Creating Games. Extracted on 27 Sep 2014 from: <http://unity3d.com/unity/download>.

Wilkis, K. (2015). Who's Watching You? Video Surveillance-User Beware. Extracted on 23 Oct 2015 from: < http://blog.willis.com/2015/05/whos-watching-you-video-surveillance-user-beware/ > .

Williams, J., and Lee, W. S. (2006). Interactive Virtual Simulation for Multiple Camera Placement. In IEEE International Workshop, Haptic Audio Visual Environments and their Applications (pp. 124-129).

Yabuta, K., and Kitazawa, H. (2008). Optimum Camera Placement Considering Camera Specification for Security Monitoring. ISCAS 2008. IEEE International Symposium on Circuits and Systems (pp. 2114-2117).

Yao, Y., Chen, C. H., Abidi, B., Page, D., Koschan, A., and Abidi, M. (2010). Can You See Me Now? Sensor Positioning for Automated and Persistent Surveillance. IEEE Transactions, Systems, Man, and Cybernetics, Part B: Cybernetics, Vol. 40(1): 101-115.

Yang, J., Arif, O., Vela, P. A., Teizer, J., and Shi, Z. (2010). Tracking Multiple Workers on Construction Sites Using Video Cameras. Advanced Engineering Informatics, 24(4), 428-434.

Yi, S. K., Haralick, R. M., and Shapiro, L. G. (1995). Optimal Sensor and Light Source Positioning for Machine Vision. Computer Vision and Image Understanding, 61(1), 122-137.

Yu, X., and Gen, M., (2010). Introduction to Evolutionary Algorithms. 1st Ed., Springer-Verlag London Limited, London.

Zhao, T., and Nevatia, R. (2004). Tracking Multiple Humans in Complex Situations. Pattern Analysis and Machine Intelligence, IEEE Transactions on, Vol. 26(9): 1208-1221.

Zheng, D., Ng, S., and Kumaraswamy, M., (2005). "Applying Pareto Ranking and Niche Formation to Genetic Algorithm-based Multiobjective Time–cost Optimization." Journal of Construction Engineering and Management, Vol. 131(1): 81-91.

APPENDICES

APPENDIX A: EXAMPLES OF SURVEILLANCE CAMERAS

- The following table includes some examples of surveillance cameras (Pelco, 2014).

Camera type		**Description**	**Some features**
Indoor fixed camera (SureVision 2.0)		Ensuring performance during missions of critical applications and used in difficult lighting conditions.	• Wide dynamic range • Advance low light performance • 3D noise filtering- • 30 image/second at 3MPx
Indoor fixed camera (Sarix Value Range)		IP network cameras that are suitable for indoor fixed video security applications. Provide high quality videos and it is cost effective.	• 30 image/second • Minimum illumination down to 0.5 lux • Integrated 68° FOV lens-web-based user interface for easy setup and configuration
Indoor PTZ camera (Sarix Professional Range)		Different types of indoor and environmental IP cameras that include the most popular features with affordable prices.	• Up to 5MPx resolution • Up to 30 image/second at 1080p • Motion detection • Operating temperature from -40° to 50°
Indoor PTZ camera (Spectra HD)		High speeds doom cameras that produce high quality images to all directions (pan, tilt and zoom directions).	• The ability to control and monitor videos via IP network • Include Auto tracker and adaptive motion detection • Wide dynamic range cameras
Outdoor PTZ camera (Esprit SE IP)		It's less susceptible to vibration that PTZ cameras. Mostly used in traffic monitoring and city centre surveillance.	• Tilt range from 36° to -85° • The ability to control video using IP network • Auto tracking • Internal scheduling clock • On screen compass and tilt display.

APPENDIX B: SCRIPT FOR GENERATING CELLS

```
// Unity3D C# script for generating the cells
using UnityEngine;
using UnityEditor;
using System.Collections;
using System.IO;
using System;

[ExecuteInEditMode]
[Serializable]
public class CellsCreatorAttachedScript : MonoBehaviour
{
public GameObject floor;
public static CellsCreatorAttachedScript instance;
public int CellsInX = 1;
public int CellsInY = 1;
public int CellsInZ = 1;
public Vector3 Dimension_cell = new Vector3 (1, 1, 1);
public Color cellsColor;
public int totalCells;
public int totalVisibleCells;
public int totalInvisibleCells;
public GameObject cellPrefab;
public GameObject TargetedFloor;
public GameObject cellsContainer;
public string TargetedFloor Name = "None";
public bool cellDisplay;
GameObject currentContainer, currentCell;
Bounds boundsOfPlane;
Vector3 sizeOfPlane, centreOfPlane;
Texture2D cellTexture;
void Awake ()
{
instance = this;
cellDisplay = true;
}
void Start ()
{
totalCells = totalVisibleCells = totalInvisibleCells = 0;
cellsContainer = (GameObject)Resources.Load
("cellsContainer");
cellPrefab = (GameObject)Resources.Load ("cell");
cellTexture = (Texture2D)Resources.Load ("cellTexture");
}

// Update is called once per frame
```

```
void Update ()
{
//CalculateTotalCells ();
}

public void Cell_Generator ()
Vector3 Area_Dimension =
Monitored_Area.GetComponent<Renderer> ().bounds.size;
CellsInX = (int) Mathf. Monitored_Area (Area_Dimension.x /
Dimension_cell.x);
CellsInZ = (int) Mathf. Monitored_Area (Area_Dimension.z /
Dimension_cell.z);
public void updateStatic () {
//iColors = importanceColors;
//iValues = importanceValues;
}
public void CreateAllCells ()
{
instance = this;
totalCells = totalVisibleCells = totalInvisibleCells = 0;
currentContainer = (GameObject)Instantiate (cellsContainer,
Monitored_Area.transform.position, Quaternion.identity);
currentContainer.transform.parent =
Monitored_Area.transform;
TargetedFloor = Monitored_Area;
floorbounds =
TargetedFloor.GetComponent<Renderer>().bounds;
floorSize = floorbounds.size;
centreOfPlane = boundsOfPlane.center;
Vector3 Position_cell;
float initx, inity, initz;
initx = centre_Floor.x - (Cells_in_X * Dimension_cell.x) /
2;
inity = centre_Floor.y;
initz = centre_Floor.z - (Cells_in_Z * Dimension_cell.z) /
2;
Material material = new Material (Shader.Find
("Transp_cells"));
material.color = cellsColor;
material.mainTexture = cellTexture;
for (int i = 0; i < Cells_in_X; i++)
for (int j = 0; j < Cells_in_Y; j++)
for (int k = 0; k < Cells_in_Z; k++) {
Position_cell = new Vector3 (initx + i * Dimension_cell.x +
Dimension_cell.x / 2, inity + j * Dimension_cell.y +
Dimension_cell.y / 2, initz + k * Dimension_cell.z +
Dimension_cell.z / 2);
```

```
currentCell = (GameObject)Instantiate (cellPrefab);
currentCell.transform.position = Position_cell;
currentCell.GetComponent<Renderer> ().material = material;
currentCell.transform.localScale = new Vector3
(Dimension_cell.x, Dimension_cell.y, Dimension_cell.z);
currentCell.transform.parent = currentContainer.transform;
currentCell.SetActive (true);
```

APPENDIX C: SCRIPT FOR CALCULATING THE COVERAGE OF FIXED CAMERAS

```
// Unity3D C# script for calculating the coverage of fixed
cameras
using UnityEngine;
using System;
using System.Collections;

[ExecuteInEditMode]
public class CellAttachedScript : MonoBehaviour {

public int importance=0;
public int importance_Index=-1;
public bool Update_required = true;
public Color invisibleColor = Color.gray;
public static Color[] FOV;
public static bool cleanedAll;
public bool[] cameras;
//Rendering the cell color during the coverage process
public static void setColors(Color[] colors )
}
CellAttachedScript. FOV = colors;
public void InsideFOV (int i)
if (importanceIndex!=-1)
ImportanceColors.present[importanceIndex]=true;
public void updateImportance()
ImportanceColors instance =
FindObjectOfType<ImportanceColors> ();
if (importanceIndex!=-1)
importance =
instance.importanceValues[importanceIndex].value;
Material material = instance.getMaterial(importance);
if (material == null)
gameObject.GetComponent<Renderer> ().sharedMaterial.color =
invisibleColor;
else
gameObject.GetComponent<Renderer> ().material = material;
void OnRenderObject()
if (!Application.isPlaying&& Update_required)
updateImportance();
Update_required = false;
// conducting the visibility process
totalCells = totalVisibleCells = totalInvisibleCells = 0;
public int calculateCells (FOV_observ, int cameraIndex)
totalVisibleCells = 0;
totalInvisibleCells = 0;
```

```
int PhyObstLayerMask = LayerMask.GetMask
("Physical_Obstacles");
if (floor.transform.childCount > 0)
totalCells = floor.transform.GetChild (0).childCount;
for (int i = 0; i < totalCells; i++)
Vector3 testPosition = floor.transform.GetChild
(0).GetChild (i).GetComponent<Renderer> ().bounds.center;
if (Tools.isVisibleFrom (observer, testPosition))
Ray ray = new Ray (observer.transform.position,
testPosition-observer.transform.position);
if (!Physics.Raycast(ray, (testPosition
observer.transform.position).magnitude, PhyObstLayerMask))
totalVisibleCells++;
floor.transform.GetChild (0).GetChild
(i).GetComponent<CellAttachedScript> (). InsideFOV
(cameraIndex);
else
totalInvisibleCells++;
else
totalCells = totalVisibleCells + totalInvisibleCells;
return totalVisibleCells;
// Conducting the camera coverage method
public static string visibility ()
float[] cameraValues = new float[CellAttachedScript.
FOV.Length];
float totalImportance=0, visibleImportance=0;
foreach (CellAttachedScript cell in
GameObject.FindObjectsOfType<CellAttachedScript>())
for (int i = 0; i<CellAttachedScript. FOV.Length; i++)
if (cell.cameras[i])
cameraValues[i]+=cell.importance;
visibleImportance+=cell.importance;
totalImportance+=cell.importance;
string output = "";
output+=String.Format("{0:0.00}\n",(visibleImportance/total
Importance)*100);
return output;

// writing the output results
CellAttachedScript.updateColorAll();
bool empty = false;
for (int i = 0; i<ImportanceColors.present.Length; i++)
if
(ImportanceColors.instance.importanceValues[i].mustBePresen
t)
if (!ImportanceColors.present[i])
empty = true;
```

```
break;
if (empty)
output.Write ("0.00\n");
else
output.Write(CellAttachedScript.visibility());
ImportanceColors.resetPresent();
output.Close();
```

APPENDIX D: SCRIPT FOR OPTIMIZING THE PLACEMENT OF FIXED CAMERAS

```
// Matlab code for optimizing the placement of fixed
cameras
// conducting the crossover

function child =
cameras_crossover_1point(parents,problem,~,population)

n_parents = length(parents);
randparents = randperm(n_parents);

parent1 = population.ind{parents(randparents(1))};
parent2 = population.ind{parents(randparents(2))};
cross_point = randi([0, problem.chromosomes], 1,1);
if cross_point == 0 || cross_point == problem.chromosomes
  if cross_point == 0 || cross_point == problem.chromosomes
    child = parent2;
  else
    child = parent1;
  end
else
    fromP1 = parent1(1:cross_point, :);
    fromP2 = parent2(cross_point+1:problem.chromosomes, :);
    child = vertcat(fromP1, fromP2);
end


end

function child =
cameras_crossover_mean(parents,~,~,population)
% Generates a child from certain parents
n_parents = length(parents);
randparents = randperm(n_parents);

parent1 = population.ind{parents(randparents(1))};
parent2 = population.ind{parents(randparents(2))};
child.x = ceil(mean([parent1.x parent2.x]));
child.y = ceil(mean([parent1.y parent2.y]));
child.z = ceil(mean([parent1.z parent2.z]));
child.pan = ceil(mean([parent1.pan parent2.pan]));
child.tilt = ceil(mean([parent1.tilt parent2.tilt]));
child.FOV = ceil(mean([parent1.FOV parent2.FOV]));

end
```

```
// conducting the mutation

function x = cameras_mutation_generic(x,problem,~,~)

chromosome1 = randi([1 problem.chromosomes], 1,1);
chromosome2 = randi([1 problem.chromosomes], 1,1);

moveX = randi([-problem.mutation_moveMax
problem.mutation_moveMax], 1,1);
moveY = randi([-problem.mutation_moveMax
problem.mutation_moveMax], 1,1);
moveZ = randi([-problem.mutation_moveMax
problem.mutation_moveMax], 1,1);
rotatetilt = randi([-problem.mutation_rotateMax
problem.mutation_rotateMax], 1,1);
rotatepan = randi([-problem.mutation_rotateMax
problem.mutation_rotateMax], 1,1);
moveFOV = randi([-problem.mutation_moveMax
problem.mutation_moveMax], 1,1);

switch randi([1 3], 1,1)
    case 1 %set to 0
        x(chromosome1, :) = zeros(1, 6);
    case 2 %change type
        backup = x(chromosome1, 6);
        x(chromosome1, 6) = x(chromosome2, 6);
        x(chromosome2, 6) = backup;
    case 3 %move roatate
        x(chromosome1, 1)= x(chromosome1, 1) + moveX;
        x(chromosome1, 2)= x(chromosome1, 2) + moveY;
        x(chromosome1, 3)= x(chromosome1, 3) + moveZ;
        x(chromosome1, 4)= x(chromosome1, 4) + rotatepan;
        x(chromosome1, 5)= x(chromosome1, 5) + rotatetilt;
   x(chromosome1, 6)= x(chromosome1, 6) + moveFOV;
        if x(chromosome1, 1) < problem.x_min
            x(chromosome1, 1) = problem.x_min;
        end
        if x(chromosome1, 1) > problem.x_max
            x(chromosome1, 1) = problem.x_max;
        end

        if x(chromosome1, 2) < problem.y_min
            x(chromosome1, 2) = problem.y_min;
        end
        if x(chromosome1, 2) > problem.y_max
            x(chromosome1, 2) = problem.y_max;
        end
```

```
        if x(chromosome1, 3) < problem.z_min
            x(chromosome1, 3) = problem.z_min;
        end
        if x(chromosome1, 3) > problem.z_max
            x(chromosome1, 3) = problem.z_max;
        end
        if x(chromosome1, 4) < problem.pan_min
            x(chromosome1, 4) = problem.pan_min;
        end
        if x(chromosome1, 4) > problem.pan_max
            x(chromosome1, 4) = problem.pan_max;
        end
        if x(chromosome1, 5) < problem.tilt_min
            x(chromosome1, 5) = problem.tilt_min;
        end
        if x(chromosome1, 5) > problem.tilt_max
            x(chromosome1, 5) = problem.tilt_max;
        end
        if x(chromosome1, 6) < problem.FOV_min
            x(chromosome1, 6) = problem.FOV_min;
        end
        if x(chromosome1, 6) > problem.FOV_max
            x(chromosome1, 6) = problem.FOV_max;
        end
end


// generating random solutions

function [ solution, settings ] = cameras_generate_random(
problem, settings )
%cameras_generate_random generates a solution for the
problem in range
%defined in problem
solution = zeros(problem.chromosomes, 6);

for i= 1:problem.chromosomes
   if settings.empty(i) < 0.3 * problem.solutions &&
randi([0 5], 1,1) == 0
    settings.empty(i) = settings.empty(i) + 1;
    continue;
   end
   [rows colls] = size(problem.typeCostTable);
   type = problem.typeCostTable(randi([1 rows], 1,1), 1);
   x = randi([problem.x_min problem.x_max], 1,1);
   y = randi([problem.y_min problem.y_max], 1,1);
   z = randi([problem.z_min problem.z_max], 1,1);
```

```
   pan = randi([problem.pan_min problem.pan_max], 1,1);
   tilt = randi([problem.tilt_min problem.tilt_max], 1,1);

   solution(i, :) = [x, y, z, pan, tilt, FOV];
end
solution
```

APPENDIX E: SCRIPT FOR CALCULATING THE CP COVERAGE OF PTZ CAMERA

```
// Unity3D C# script for calculating the coverage of PTZ
cameras
using UnityEngine;
using System;
using System.Collections;
using System.Text;

[ExecuteInEditMode]
public class CellAttachedScript : MonoBehaviour {

public int importance=0;
public int importanceIndex=-1;
public bool needsUpdate = true;

public Color invisibleColor = Color.gray;
public static Color[] cameraColors;
public static bool cleanedAll;
public bool[] cameras;
public static float onePositionVisibility;

public static void setColors(Color[] colors ) {
CellAttachedScript.cameraColors = colors;
}
public void setVisibleFromCamera (int i) {
      if (importanceIndex!=-1) {
      ImportanceColors.present[importanceIndex]=true;
        }
      if (cleanedAll) cleanedAll = false;
      if (i>=0 && i<cameraColors.Length) cameras[i] = true;
      }

public void reset() {
      if (cameraColors!=null) {
      cameras = new bool[cameraColors.Length];
      }
}

public static void resetAll() {
      if (cleanedAll) return;

      foreach (CellAttachedScript cell in
GameObject.FindObjectsOfType<CellAttachedScript>()) {
            cell.reset();
      }
```

```
     cleanedAll = true;
}

public static void updateColorAll() {

     foreach (CellAttachedScript cell in
GameObject.FindObjectsOfType<CellAttachedScript>()) {
          cell.updateColor();
     }
}

public void updateImportance() {
     ImportanceColors instance =
FindObjectOfType<ImportanceColors>();
     if (importanceIndex!=-1) {
     importance =
instance.importanceValues[importanceIndex].value;
     }
     Material material = instance.getMaterial(importance);
     if (material == null) {
     gameObject.GetComponent<Renderer>().sharedMaterial.col
or = invisibleColor;
     } else {
     gameObject.GetComponent<Renderer>().material =
material;
     }
     }

void OnRenderObject() {
     if (!Application.isPlaying&&needsUpdate){
     updateImportance();
     needsUpdate = false;
     }
     }

// conducting the visibility process

public static float[]/*string*/ visibility()
       {
       float[] output = new float[2];
     float[] cameraValues = new
float[CellAttachedScript.cameraColors.Length];
     float totalImportance=0, visibleImportance=0;
foreach (CellAttachedScript cell in
GameObject.FindObjectsOfType<CellAttachedScript>()) {
```

```
     for (int i = 0;
i<CellAttachedScript.cameraColors.Length; i++){
     if (cell.cameras[i]) {
     cameraValues[i]+=cell.importance;
     visibleImportance+=cell.importance;
     }
     totalImportance+=cell.importance;
     }
     }
       TextIO textio = FindObjectOfType<TextIO>();
int PhaseTime =
textio.CamerasStorage[textio.camerasStorageIt].Checkpoints[
textio.record].Priv;
        onePositionVisibility = PhaseTime *
visibleImportance;
        output[0] = onePositionVisibility;
        output[1] = (float)totalImportance;
     return output;
     }

void Start() {
     this.invisibleColor =
CellsCreatorAttachedScript.instance.cellsColor;
     }
     void updateColor () {
float r, g, b, a, v;
     v=r=g=b=a=0;
for (int i = 0; i<cameras.Length; i++) {
     if (cameras[i]) {
     r+=cameraColors[i].r;
     g+=cameraColors[i].g;
     b+=cameraColors[i].b;
     a+=cameraColors[i].a;
     v+=1;
     }
     }
v*=(float)Math.Pow(0.7, v-1);
     r=Mathf.Clamp(r/v, 0f, 1f);
     g=Mathf.Clamp(g/v, 0f, 1f);
     b=Mathf.Clamp(b/v, 0f, 1f);
     a=Mathf.Clamp(a/v, 0f, 1f);

     if (v!=0) {
     gameObject.GetComponent<Renderer>().material.color =
new Color (r, g, b, a);
     } else {
```

```
     gameObject.GetComponent<Renderer>().material.color =
invisibleColor;
     }
     }
void OnDestroy() {
     }
}
void CalculateAndFormatAndWrite(float[]
onePosVisAndTotalImportance)
      {
// calculating the CP coverage

  bool recorded = false;
       float visibilityOfAnyLine = 0;
       if (recorded == false)
       {
       for (int rec = 0; rec <
recordsIndexesLists[currentLine].Count; rec++)
       {
       if (recordsIndexesLists[currentLine][rec] == record)
       {
       recordsPositionsVisibilityList[currentLine][rec] =
onePosVisAndTotalImportance[0];
       if (rec == recordsIndexesLists[currentLine].Count -
1)
       {
       float summationOfAllPositionsVisibilityInLine = 0;
       for (int i = 0; i <
recordsPositionsVisibilityList[currentLine].Count; i++)
       {
summationOfAllPositionsVisibilityInLine +=
recordsPositionsVisibilityList[currentLine][i];
       }
Debug.Log("<b><color=blue>summationOfAllPositionsVisibility
InLine  </color></b><b><color=brown>" + currentLine +
"</color></b> = " +
debugPrintListLine(recordsPositionsVisibilityList,
currentLine) + " = <b><color=blue>" +
summationOfAllPositionsVisibilityInLine +
"</color></b>\n\t");
visibilityOfAnyLine =
(summationOfAllPositionsVisibilityInLine /
(onePosVisAndTotalImportance[1] * CY)) * 100;
string eqStr =
"<b><color=green>visibilityOfAnyLine</color></b> =
(summationOfAllPositionsVisibilityInLine / (TotalImportance
* CY))*100 =\n";
```

```
Debug.Log(eqStr + "\t\t= (\t\t" +
summationOfAllPositionsVisibilityInLine + "\t\t / " + "(
" + onePosVisAndTotalImportance[1] + "          *   " + CY +
"))*100 = " + "<b><color=green>" + visibilityOfAnyLine +
"</color></b>");
       string result;
       result = String.Format("{0:0.00}",
visibilityOfAnyLine);
       if (record < numberOfRecords-1) output.Write(result
+ "\n");
 else if (record == numberOfRecords-
1)output.Write(result.ToString().TrimEnd('\r',    '\n'));
        recorded = true;
        currentLine++;
        }
        }
        if (recorded == true) break;
        }
       }
    }
```

APPENDIX F: SCRIPT FOR APPLYING GA TO OPTIMIZE THE PLACEMENT OF PTZ CAMERA

```
// Matlab code for optimizing the placement of PTZ cameras
// generating the time in every solution

Function[dcvr]=sltnFcn(domangle,fov,T,minTP,maxTP,minTt,max
Tt)

Tt = randi([minTt,maxTt],1,1);
K = int(domangle/FOV);
Total = T-Tt*(K);
Flag = true;
while flag
    mTP = randi([min TP,max TP],1, K);
    mTP (1,K)= total-sum(mTP (1,1: K-1));
    if mTP(1,K)>= minTP && mTP(1,K)<= maxTP
        flag = false;
    end
end


// generating random solutions

function [pop,pop3,pop1,sec]=ptzpopFcn(parameter)

ndcvr=parameter(1,1);
domangle=parameter(10,1);
fov=parameter(11,1);
sp=int16(domangle/fov);
T=parameter(2,1);
minTP=parameter(3,1);
maxTP=parameter(4,1);
minTt=parameter(5,1);
maxTt=parameter(6,1);
for i=1:k
    minr(i)=parameter(11+i*2-1,1);
    maxr(i)=parameter(11+i*2,1);
end

pop=cell(ndcvr,1);
for popm=1:ndcvr

pop{popm,1}=sltnFcn(angle,v,q,minp,maxp,mins,maxs,minr,maxr
);
end
```

```
sec=zeros(ndcvr,1);
for secm=1:ndcvr
    sec(secm,1)=size(pop{secm,1},2);
end
maxsec=max(sec);
pop1=zeros(ndcvr,maxsec);
for pop1m=1:ndcvr
    pop1(pop1m,1:sec(pop1m,1))=pop{pop1m,1};
end
pop2=zeros(ndcvr,maxsec/6);
for pop2m=1:ndcvr
    for pop2n=1:maxsec/6
        pop2(pop2m,pop2n)=sum(pop1(pop2m,((pop2n*6)-
5):(pop2n*6)));
    end
end
pop3=cell(ndcvr,maxsec/6);
for pop3m=1:ndcvr
    for pop3n=1:maxsec/6
        if pop2(pop3m,pop3n)==0
            pop3{pop3m,pop3n}=[];
        else
            pop3{pop3m,pop3n}=pop1(pop3m,((pop3n*6)-
5):(pop3n*6));
        end
    end
end
```

APPENDIX G: SAMPLE OF TEXT FILES FROM THE PROCESS OF OPTIMIZING THE PLACEMENT OF MULTIPLE CAMERAS

- Every line in the table is a solution with four types of cameras.
- Each line in a text file includes the variables: X, Y, Z, Pan, Tilt, FOV angles.
- If the line includes (0 0 0 0 0 0), it means the camera is not joining this solution (the number of cameras is changing in every solution).

TEXT FILE 1	TEXT FILE 2	TEXT FILE 3	TEXT FILE 4
12 100 63 100 7 100	97 100 98 175 25 100	0 0 0 0 0 0	0 0 0 0 0 0
71 100 27 35 5 60	32 100 3 137 14 92	18 100 45 256 18 100	0 0 0 0 0 0
59 100 75 182 10 80	0 0 0 0 0 0	82 100 93 70 12 80	47 100 83 198 17 92
76 100 38 27 16 80	78 100 13 169 16 92	0 0 0 0 0 0	66 100 75 30 14 80
15 100 54 28 25 100	97 100 78 313 22 60	26 100 43 65 24 80	13 100 58 52 16 60
35 100 40 86 6 92	0 0 0 0 0 0	34 100 37 281 7 80	40 100 13 345 24 60
0 0 0 0 0 0	73 100 45 106 16 92	69 100 37 281 18 60	78 100 44 110 14 100
82 100 65 292 12 92	94 100 55 211 18 80	47 100 85 81 9 80	44 100 93 66 14 60
44 100 26 214 13 100	0 0 0 0 0 0	0 0 0 0 0 0	58 100 46 197 25 80
0 0 0 0 0 0	89 100 80 94 7 60	13 100 10 178 18 92	0 0 0 0 0 0
.	.	.	.
.	.	.	.
.	.	.	.

APPENDIX H: SAMPLE OF OUTPUT TEXT FILES (COVERAGE RESULTS OF ONE GENERATION) FROM THE GAME ENGINE

- The table shows 10 coverage results of one generation.
- The coverage results represented as a percentage.

Solution #	Coverage results
1	7.60
2	11.71
3	11.16
4	5.06
5	13.11
6	13.56
7	11.88
8	10.96
9	8.51
10	15.64
.	.
.	.
.	.

APPENDIX I: SAMPLE OF TEXT FILE (ONE GENERATION) FROM THE PROCESS OF OPTIMIZING THE PLACEMENT OF PTZ CAMERA

- Every line is a solution with four phases. Each phase includes: X, Y, Z, pan, tilt and the phase time.

Text file component			
49 100 14 19 21 24	49 100 14 126 32 7	49 100 14 210 6 24	49 100 14 309 66 23
26 100 13 39 81 20	26 100 13 115 48 13	26 100 13 219 32 23	26 100 13 286 71 22
6 100 35 37 1 7	6 100 35 91 30 22	6 100 35 209 14 24	6 100 35 299 81 25
38 100 53 16 50 14	38 100 53 130 37 21	38 100 53 208 46 20	38 100 53 279 54 23
26 100 68 6 79 22	26 100 68 94 20 23	26 100 68 202 65 21	26 100 68 302 75 12
58 100 24 40 5 16	58 100 24 112 30 16	58 100 24 225 90 22	58 100 24 293 85 24
40 100 42 30 43 12	40 100 42 103 78 21	40 100 42 180 69 20	40 100 42 277 19 25
37 100 99 7 15 16	37 100 99 119 68 21	37 100 99 188 77 18	37 100 99 292 21 23
10 100 88 37 47 22	10 100 88 117 4 20	10 100 88 199 56 12	10 100 88 277 32 24
32 100 61 35 76 19	32 100 61 94 48 20	32 100 61 187 57 25	32 100 61 290 83 14
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.