BIM BASED OPTIMIZATION OF CCTV SURVEILLANCE SYSTEM IN BUILDINGS, A CASE STUDY OF HOSPITALS

FINAL YEAR PROJECT UG-2019

By

Taha Aziz (G.L)	284513
Mehmood Ahmed	290168
M. Elaf Hassan	283087
Hamza Parwez	288863

(UG BECE 2019 NICE SCEE NUST)

Project Advisor

Dr. Muhammad Umer Zubair

NUST Institute of Civil Engineering School of Civil and Environmental Engineering National University of Sciences and Technology

Islamabad, Pakistan

2023

This is to certify that the Final Year Project

Bim based optimization of cctv surveillance system in buildings, A case study of hospitals

Submitted by

Taha Aziz (G.L)	284513
Mehmood Ahmed	290168
M. Elaf Hassan	283087
Hamza Parwez	288863

Has been accepted towards the requirements

for the award of undergraduate degree.

Bachelor of Engineering in Civil Engineering

Dr. Muhammad Umer Zubair

Assistant Professor

Construction Engineering and Management Department

NUST Institute of Civil Engineering (NICE)

School of Civil and Environmental Engineering (SCEE)

National University of Sciences and Technology, Islamabad, Pakistan

Acknowledgement

Commend be to All-powerful ALLAH, Most Thoughtful, Most Kind, for presenting us with the knowledge and strength to complete our final thesis. We would like to thank our advisor, Dr. Muhammad Umar Zubair for giving us the opportunity to work beneath his supervision and guidance. It has been an honor for us to have such a qualified and distinguished personality to administer us all through our investigative stage, and we are incapable to describe in words our appreciation towards our tutor for contributing his valuable time and experience, completing this project would never have been conceivable if we never had the steady backing, assistance and the thrust to move forward from our advisor. We are moreover thankful to Dr. Muhammad Usman for providing us with plans that facilitated in the conduct of our FYP. We are also thankful to all the previous researchers from whose papers we could seek knowledge and guidance. We would also like to recognize the positive state of mind of our classmates towards our work which always gave us the motivation to move forward and make our project worthy of being presented. Finally, and most importantly, we would like to thank the foremost vital individuals in our lives, our guardians and siblings, for supporting us through all our work and endeavors.

Abstract

CCTV surveillance is essential in guaranteeing the safety and security of buildings, particularly big and complex structures. The existing manual layout design approach is subjective, prone to human mistake, and lacks exact visualization of camera views, resulting in blind areas and excessive camera overlap. This study provides a novel way to optimize camera location utilizing GA and BIM. This research is divided into three sections. First, The data extraction module extracts relevant information from the BIM model using pyRevit. Second, the optimization framework, written in Python, uses GA to optimize camera placement depending upon user-defined parameters such as number of cameras or desired coverage percentage. Third, the visualization module uses pyRevit to incorporate the optimized camera placements and orientation back into the BIM model, allowing for a visual depiction of the camera configuration. A case study was conducted on the ground floor lobby of River Garden Hospital to validate the suggested framework. The manually designed camera layout was compared to the optimum layout obtained by the suggested framework. The results showed that the optimized arrangement outperformed the manual design in terms of coverage percentages and overlapping regions. Furthermore, optimizing the number of cameras resulted in improved coverage with a smaller number of cameras

Table of Contents

LIST OF ACRONYMS

BIM	Building Information Modeling
CAD	Computer-aided design
GA	Genetic Algorithm
CCTV	Closed Circuit Television
FOV	Field of View
PDF	Portable Document Format
AGP	Art Gallery Problem
AEC	Architecture, Engineering and Construction
API	Application Programming Interface
2D	Two Dimensional
3D	Three Dimensional

LIST OF FIGURES

Chapter 1: Introduction

Section 1.1: Background

Closed Circuit Television (CCTV) surveillance is now a days a fundamental part of the design process of new and especially large buildings. A CCTV layout indicates the position of individual cameras and their respective orientation and Field of View (FOV) in real time. Every building has certain areas which require constant monitoring. These areas are termed as regions of interest (ROIs). While designing a camera network layout, it is important to take into consideration these ROIs and technical specifications of individual cameras to obtain maximum coverage. In practice, client specific needs, such as budget, number of cameras or percentage of coverage required, also affect the arrangement of cameras [1].

Currently in the industry, surveillance systems or camera networks are designed manually by experts based on 2D plans and site visits of the relevant building, however the manual placement has certain flaws. First, the surveillance system is designed by experts based on their personal experience and judgement, which makes the surveillance layout highly subjective. Second, the lack of precise visualization of cameras' FOV results in several blind spots. These blind spots are later identified in the view of installed cameras and the relocation process is expensive and time consuming [3]. Third, more cameras are provided to increase the coverage area resulting in greater overlap region and subsequently, higher cost.

Section 1.2: Problem Statement

To overcome the above-mentioned flaws, there is a need to implement an optimization algorithm that develops a surveillance camera network by maximizing the coverage with minimum number of cameras, reducing the overlap and blind spots. This will also eliminate the human error associated with manual design of surveillance system. For precise visualization of the FOV of individual cameras in the network, Building Information Modelling (BIM) can be used.

BIM has been widely adopted in the construction industry in recent decades however its integration with CCTV is very limited [4]. The usage of BIM in development of camera networks can improve its performance and help identify various elements that surround the coverage area [5]. BIM enables us to visualize cameras' view before its actual installation. For this, a virtual 3D model of building is required in any BIM based software. Revit was the best suited BIM software for the scope of this study. It was chosen for its simplicity of exporting building information such as coordinates in a separate file. Revit also has a built-in camera tool making the visualization of camera's view simple and easy.

Optimization of camera coverage is a complex problem requiring many calculations and iterations. This can be done effectively by using Genetic Algorithm (GA). It is a search-based optimization algorithm based on the principle of genetics and natural selection. GA has proven to be highly effective for dealing with large search spaces and reducing their optimization time [6]. Optimizing the location of multiple cameras in a network and their respective orientation generates a significantly large solution space making this problem difficult to be solved by traditional methods, hence GA is used to reduce time and providing better convergence towards the most optimal solution.

This study aims to optimize camera placement based on Genetic Algorithm (GA) and user defined data. The user can input the number of cameras as per their budget and find their most optimal position within the region, or the user can input the percentage coverage and find the minimum number of cameras required and their respective positions to achieve desired coverage, while keeping into consideration the obstacles and region of interests (ROI), present in the optimization area. The field of view and range of each individual camera can also be customized making the optimization process of a camera network like actual site conditions.

Section 1.3: Objectives

This study has following goals:

1. To develop a framework for exporting building data from BIM model and implementing results back on it.
2. Develop an optimized CCTV layout using algorithms considering relevant constraints.
3. Application of results on the case study of a hospital.

Section 1.4: Scope and Limitations

Every study has its restrictions and limitations. In our study they are:

1. Pan and Tilt of the cameras not considered: The project's limitation of not considering pan and tilt movements of the cameras somewhat restricts its ability to optimize camera placement, potentially overlooking the optimal coverage and angles required in real-world scenarios.

2. Limited to 2D space: By confining the optimization process to a 2D grid, the project fails to account for the vertical dimension, neglecting crucial factors such as varying heights of objects and their impact on camera placement, which could affect the overall effectiveness and accuracy of the optimized camera positions.

3. Resource Intensive approach: The project's utilization of a grid-based representation for the search space proves to be resource-intensive, particularly when scaling up the number of cameras or employing high-resolution grids. This can lead to longer computation times and increased computational resource requirements, impeding real-time or large-scale deployment scenarios.

Section 1.5: Research significance

The contribution of this study includes.

1. Integration of BIM and CCTV

2. Application of GA to optimize the placement of camera network, considering relevant constraints.

3. Development of a cheaper and efficient method to design a camera layout reducing the cost, site visits, number of cameras and human error associated with manual design of a surveillance system, while providing the most efficient coverage.

Section 1.6: Thesis Structure

This document is divided into five chapters:

1. Started with providing the context of the study area, problem statement, outlining the research objectives, defining its scope and limitations, and research significance in Chapter 1.

2. Chapter 2 provides an in depth and thorough review of the literature. The chapter also outlines research holes in this field.

3. Chapter three explains the methodology framework that was adopted to implement the project.

4. Chapter four showcases the results of the project and provides a detailed discussion on the outcomes.

5. Finally, chapter five concludes the project and offers recommendations for future research.

Chapter 2: Literature Review

Section 2.1: Traditional Methods

The integration of BIM and CCTV is an area that has received limited attention in the existing literature. However, studies have been conducted on the optimization of coverage and automatic camera placement. The Oldest coverage optimization is the Art Gallery problem (APG) [7]. It involves measuring the position and minimum number of guards to fully supervise important areas in a polygon. Guards do not have any view restrictions as compared to the restricted FOV of cameras so solutions of AGP cannot be directly applied to camera networks. To solve this, Horster and Lienhart [8]. proposed a liner programming model. They assumed the polygon to be a simple rectangle and did not consider the presence of any obstructions or constraints inside the polygon.

Section 2.2: Introduction to ROIs

Yabuta and Kitazawa [9]. proposed a segmentation approach for the automatic placement of cameras. Their approach involved dividing the polygon into rectangular segments allowing for coverage in irregular shaped polygons. They suggested that for effective monitoring only selective areas need to be observed through cameras, they named those selective areas as ‘essential regions. To the best of our knowledge, these were the first ones to introduce the concept of region of interests in the problem of optimizing camera coverage. However, their approach did not consider rest of the constraints and could not be generalized to other scenarios.

Section 2.3: Usage of Optimization Algorithms

Xu and Lei (2009) [10] studied the optimization of a camera layout's field of view using Particle Swarm Optimization. In their research a fixed number of cameras were initially distributed across an area and then their respective positions were fixed, and no movement was allowed. As a result, the optimization process was considered solely on enhancing the orientation of these cameras.

Their work was further extended by Xu et al. [11]. by incorporating the position of cameras in the optimization process. They assumed every camera to be installed on wheels allowing it to move in space. After defining constraints, they applied following three operators in the particle's movement *penalty, absorbing and reflecting* for the optimization process. Even though their study is close to our problem definition, they had oversimplified assumptions for a general Camera Placement Problem. Their approach only considers rectangular surveillance area and assumes that all cameras are of the same type i.e., fixed FOV and range. Also, their approach to manage the coverage in presence of obstructions in the area was unclear and vague which made their method not applicable to ours.

Unlike previous mentioned studies focusing on minimizing overlap FOV, Yao et al. [12]. suggested that in certain scenarios it is necessary to have uniform overlap between camera FOVs. One such application of their study is object tracking. To maintain trajectory of a moving object across different cameras view it is necessary to have uniform overlap between adjacent cameras for automated and successful handover.

Section 2.4: Similar optimization problems

Zhang et al [14]. proposed a divide-and-conquer scan planning method which was based on pre-defined point goals to achieve minimum data collection time, which was able to greatly reduce the computation time compared to previous approaches. Kang et al [15] optimized the method of censor placement using partheno-genetic algorithm. Han et al [13]. optimized and analyzed camera coverage by maximal Coverage Location Problem-Complementary Coverage (MCLP-CC) model. Their approach effectively tackles the presence of Region of Interests and obstructions but is based on basic assumption that all cameras used have fixed specifications making their approach very challenging to be used in practical real-life situations.

Section 2.5: Constraints

By identifying the location of different constraints on the placement areas we can provide valid coordinates of camera placement to the optimization algorithm. This will reduce the time of optimization process by making the camera placement restricted to valid positions and avoiding the positions of constraints. Following are the constraints common for all building:

- Geometrical Constraints (signboards, columns etc)
- Logical Constraints (Positions in which camera is directly facing the walls)
- Legal Constraints (e.g in case if the monitoring area includes privacy areas)

Given below are the constraints associated with health care buildings. These are not to be covered by CCTV:

- Doctor Chambers
- Examination Room
- Restrooms
- Changing Room

By putting these constraints, we guide the algorithm to search through the arrangements that prevent coverage of these areas.

Section 2.6: BIM in Construction Industry

Construction projects are challenging to plan and execute due to technical, organizational, and management issues. Traditional methods of design-bid-build procurement and non-integrated software can result in low productivity, delays and cost overruns. Since all information is not integrated on a single platform there will be communication gaps which will make it difficult to incorporate any chances to the project this can lead to discrepancies in the project coordination which will lead to low productivity, delays or even cost overruns (Aibinu & Venkatesh, 2014).

BIM is a major technological advancement in the construction industry that offers a wide range of functionalities and is gaining acceptance due to its ability to address various issues in project delivery BIM facilitates integration and collaboration, which can lead to reduced project duration and cost. BIM is an intelligent model-based process that provides insight for creating and managing building and infrastructure projects efficiently, economically and with less environmental impact. BIM stores all project information in a single digital model, which reduces design errors and omissions and significantly reduces design time. The enriched data model provides detailed information for each project element, in addition to its use as a 3D visualization tool. (McCuen,2009).

Autodesk Revit is considered as one of the most important and widely used tools of BIM. Revit is an efficient Building Information Modeling (BIM) software used to model shapes, structures, and systems in 3D with parametric accuracy, precision, and ease. Revit drives efficient BIM workflows and includes specialized tools and automation for every AEC discipline, with tools for sketching, scheduling, annotating, and document production. Some of the uses of Revit are described in detail below.

1. Better Collaboration: Autodesk Revit allows multiple project stakeholders to collaborate seamlessly in real-time, which ensures that every concerned entity is on the same page.

2. Improved Productivity: Revit's parametric design tools and building elements library help reduce design time and increases productivity, as users can easily generate and modify building elements.

3. Enhanced Visualization: Revit's 3D modelling and rendering capabilities allow users to create highly accurate and realistic building models, enabling clients to better visualize the final product.

4. Improved Project Delivery: Revit's BIM capabilities help identify and resolve design issues early in the project lifecycle, resulting in fewer delays and cost

overruns. Additionally Revit's integration with other Autodesk products such as Dynamo, Navisworks and BIM 360, streamlines project delivery and helps ensure a successful outcome.

Autodesk Dynamo is an open-source graphical programming environment for computational BIM design that enables one to use visual logic to design workflows and automate tasks. Dynamo consists of various nodes, each having a specific function of its own. By connecting different 'nodes' with each other using 'wires', to indicate the logical flow of the resulting visual program, Dynamo allows us to develop programs that perform the desired tasks. Dynamo allows users to create systematic relationships to manipulate model elements and parameters in ways that would be impossible with traditional Revit tools.

Section 2.7: Research Gap

Most of the studies on this topic do not address both constraints and ROIs while those that do incorporate both aspects lack user customization of individual cameras making the practical implementation of these studies very difficult. Hence there is a need for a single BIM based framework that optimizes and automates the placement of camera network layout while considering relevant constraints and ROIs and then provides visualization of cameras view before installation.

Chapter 3: Methodology

This study proposes a novel approach to optimize placement of camera network in buildings. The proposed methodology has a systematic approach consisting of three key components. Fig(1). (1) Module A: Data extraction (2) Module B: Optimization framework (3) Module C: Visualization of optimized camera placement. In data extraction, information is extracted from the BIM model using custom extension after which it is filtered to relevant data. Next, optimization framework is implemented on the relevant data. Implementation uses GA to obtain optimized placements along with their respective coverage and overlap. Lastly, optimal placements are visualized back in the BIM Model of the building using another custom extension. The details of the three key components in the proposed methodology are demonstrated in the following sections.

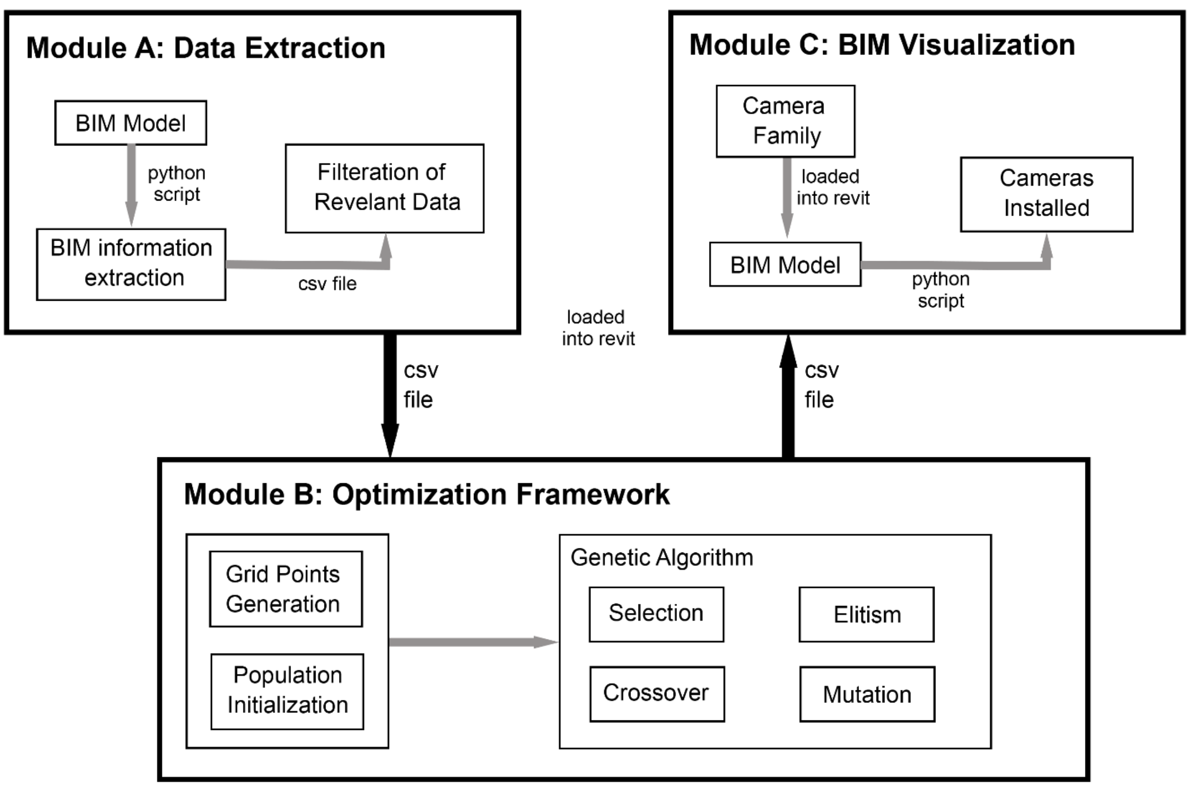

Figure 1 Methodology flowchart

Section 3.1: Module A - Data Extraction

The data extraction module involves the retrieval of data from BIM through the implementation of python plugin. BIM model represents the accurate and comprehensive information about the building including valuable information about its components and attributes. Through the implementation of Python plugin necessary information is efficiently extracted from the BIM Model, serving as basis for the subsequent stages of the framework for analysis and further processing. The purpose of Module A is to produce significant input for Module B. Module A has three major components: BIM Development, information extraction from BIM and Filtering of relevant data.

Section 3.1.1: BIM Development

Accurate extraction of data is important to yield feasible results. Due to precision and other numerous elements, BIM is an efficient method to represent accurate and detailed information about the building in a virtual environment. Before data is extracted, an accurate model of the building needs to be constructed. For this purpose, this study used Autodesk Revit for BIM modelling of the building. Autodesk Revit offers a wide range of add-ons and extensions that enhance its functionality; Therefore, it was the most suitable for implementation in this study.

Section 3.1.2: Information Extraction from BIM

Once we have the BIM model, the next step is extracting information from the model. For this purpose, an Autodesk Revit add-on pyRevit is used to gain access and manipulate the Revit's API. The information extracted will include the coordinates of the room and doors. Room where surveillance is required will be obtained from this exported data and the remaining rooms on that specific level will be considered as obstacles. The coordinates of the doors will play an important role in determining the Region of Interest (ROI) within the room. The ROI includes specific points and areas that require constant surveillance, regardless of other factors. Obtained door coordinates will help in establishing the ROI region, assuring their surveillance which is a significant aspect of the surveillance network. To obtain these coordinates an extension utilizing the python code is created. Room boundary segments coordinates are retrieved, and X Y coordinates of the segments are appended to the room coordinates. The data is exported in an organized format which allows for easy manipulation of the collected information. The extracted data, including room and door coordinates, is exported to a spreadsheet for further analysis.

Section 3.1.3: Filtering of Relevant Data

The exported spreadsheet contains coordinates of all the rooms and doors; Therefore, python script is created to filter the exported spreadsheet to contain only the necessary data. Script prompts the user to define the level at which they intend to install cameras. This ensures that the extracted information is tailored to their requirements. All rooms in that level are then displayed and the user is then prompted to enter the name of the room they are interested in. However, it's important to note that certain case-specific constraints are implemented to ensure that only appropriate rooms are selected. If they enter a room name that is within these

restrictions, the user is informed that the room is not permitted and is asked to input an alternative room name After the user chooses a valid room, the script moves on to show the doors and prompt the user to define regions of interest (ROI). The script makes sure that the information being collected is in line with the user's requirements and follows the use case's unique limitations. On completion, the relevant information is exported and stored into a spreadsheet. This information will be used as input for Module B.

Section 3.2: Module B - Optimization framework

Module B consists of 4 major components: Variable selection, search space, execution sequence, objective function, and optimization algorithm. This module makes use of python and performs all the main steps required to reach an optimal camera placement within the model.

Section 3.2.1: Variable Selection

The variables of camera placement include the position and orientation of camera. X, Y represents the position of camera and Z (Height) will not be considered in the optimization process. Camera orientation is relative to the x-axis inside a cartesian plane.

Section 3.2.2: Search Space

Before optimizing camera placement, a search space must be defined. In our proposed optimization, the search space is defined by a set of grid points that are uniformly generated within the boundaries of the room. The number of grid points along x and y axes affects the accuracy of the solution. A higher number corresponds to better optimization results and longer computational time. Grid points are generated based on the minimum and maximum x, y values of the room and it is unavoidable that some points are generated outside the defined boundaries of the room. To address this issue and ensure precise camera placement, a robust algorithm known as 'Point in Polygon' is utilized. Point in Polygon algorithm analyzes the geometric relationship between the point and the polygon's edges, counting the number of intersections between them. This algorithm efficiently identifies and removes cameras that lie outside the designated room boundary, thereby optimizing the accuracy and integrity of the grid point generation process. Grid points inside the obstacles are again removed using the point in polygon algorithm. This algorithm will again be utilized multiple times further down the module. [16]

Section 3.2.3: Execution Sequence

Two options are available at the start, either specify the number of cameras or specify the camera coverage percentage. In the first option, the total number of cameras available is entered along with their respective FOVs. If the second option is selected, the code begins with a single camera and continues iterating through cameras until the desired coverage is achieved. In both scenarios, the execution sequence required for optimized results is the same.

Section 3.2.4: Objective Function

$$\boldsymbol{maximize\ f(individial) = \frac{coverage + roi_{coverage} - overlap\ penalty * overlap}{total\ grid\ points\ within\ a\ room}}$$

individual = list of tuples, each tuple representing a camera position and orientation x, y, theta, coverage = total number of grid points within the room covered by the cameras , $\mathrm{roi}_{\mathrm{coverage}}$ = total number of grid points within the regions of interest covered by the cameras., overlap= total number of grid points within the room covered by more than one camera, overlap penalty= constant value that penalizes overlapping grid points between cameras, Total grid points within room=total number of grid points within the room, excluding any points inside the obstacles.

Grid points are included in coverage based on the following criteria:

$$1. Distance = \sqrt{dx^2 - dy^2}$$

Calculates distance between a camera and a grid point.

If Distance<Range, Criteria satisfies.

$$2. Angle = \tan^{-1}\frac{dy}{dx} - \theta$$

Calculates difference in angle of line segment (camera, grid point) and orientation of camera.

If $\frac{-CAMERA\ FOV}{2} < Angle < \frac{CAMERA\ FOV}{2}$, Criteria satisfies.

3.Blocked by obstacle

$L(t) = C + t * (G - C)$ where L(t) =Line Segment between camera & grid point

$E_{i(s)} = p_i + s * (p_{i+1} - p_i)$ where E= Edge of obstacle polygon

$t, s \in [0,1]$ such that L(t) = E_i(s), then L intersects the Obstacle

Else Criteria Satisfies

If all three criteria are satisfied, the point is included inside the camera coverage. This process is repeated for each individual grid point and coverage of the camera is calculated. Similarly, the region of interest and overlap coverage are calculated. Grid points covered by multiple cameras are included in the overlap coverage.

Section 3.2.5: Optimization Algorithm

Our study adopts Genetic Algorithm (GA) to address the optimization problem. GA is a global search technique that is based on the principle of natural selection and natural genetics, appropriate for solving our problem.GA starts initially with generating an initial set of solution called population. Individuals inside the population are called chromosomes and individuals inside the chromosome are labeled as genes. Previously defined objective function is used to calculate chromosome's fitness values and evaluation is performed based on this fitness value. Once population has been initialized, genetic operations like selection, elitism, crossover, and mutation are iterated on this population to converge the solution to an optimized placement. Certain parameters like number of generations, population size, mutation rate and elitism rate are selected through testing.

In our optimization problem, parameters are set to the following values: Number of generations= 100, population size=500, mutation rate=0.3 and elitism rate=0.02. Increasing generations and population will produce more optimized results at the cost of processing time. Overlap penalty is introduced in the fitness calculation to discourage overlap between cameras and its value is set to 0.5. Integer representation of GA is used in our study. Genetic algorithm starts with generating the initial population, random positions ranging between minimum maximum x y and orientations ranging [-180,180] with respect to x-axis are generated to ensure diverse population. Point in polygon algorithm is utilized to make sure no camera is generated inside the obstacle and outside the room boundaries. (Fig) shows the chromosome. Selection operation is performed on the population in which three chromosomes are selected from the population and one with highest fitness is selected. This process is iterated for $\frac{population}{2}$, which is 250 times in our case. Elitism is performed on the selected chromosomes to preserve and pass the best solutions from the current generation onto the next one without performing other genetic operations on it. Single point crossover is performed on the selection population which involves selecting two parents/chromosomes and swapping them at a random point. Children generated in this process are passed on for mutation. In mutation, a random value is generated between 0 to 1 and is compared with the mutation rate. If it is lesser than the mutation rate, then random changes are made to the child's position and orientation. Range for mutation: Position [-5.5] and for orientation [-30,30]. Point in polygon utilized here to ensure changes in position doesn't place the camera outside the room or inside an obstacle.

Whole cycle is repeated for the specified number of generations and after the last generation, the best chromosome is selected from the population which is essentially the optimal solution. Optimal solution is stored on a spreadsheet and will be accessed in Module C.

Section 3.3: Module C - Visualization of Optimized cameras

The purpose of this module is to visualize the optimized camera placement on to the BIM model of the building. Autodesk Revit add-on pyRevit is utilized again here to import the results back into the model. Before the code snippet is run, a camera family needs to be loaded into the project and such family is loaded from a free open-source website. Once the family component is loaded into the project, the code snippet is modified to include the family name. Custom extension pushbutton is pressed which prompts to select the level and upon selection, position and orientation are imported from the Module B output spreadsheet. Camera component is added on the exact coordinates and then rotated using the orientation from the csv file. Camera height is specified in the code snippet which can be modified to ensure placement at ceiling level.

Chapter 4: Validation

Section 4.1: Case study of a hospital

To validate the proposed framework of optimized camera placement a case study was conducted on River Garden Hospital. Surveillance system in healthcare facilities has grown increasingly necessary in recent years. Unfortunately, crimes against people and property do occur in hospitals. For example, patients are abused, infants are abducted from nurseries, patient's valuables are stolen, drugs and medical supplies are stolen, lost or misused, equipment and facilities are vandalized; and violent behavior against hospital staff occurs in emergency rooms [2]. This requires a well-established surveillance system for the safety and wellbeing of patients, staff, and visitors.

Section 4.2: Validation Experiments

Validation experiments were carried out on Ground floor lobby of the hospital. This particular area was selected based on its complex and irregular shape, presenting a unique set of challenges. It has several corridors and comprises of both narrow and wide spaces making it suitable to compare the results of manual and optimized camera placement.

For manual placement of the camera layout, a professional having over 11 years of Industrial experience in camera layout design, was consulted to design a manual layout using 2D drawings and 3D model of the building. Fig (2) shows the 2D plan of ground floor of the hospital. There may be some general constraints lying in the coverage area like restrooms and changing rooms. Also, in healthcare facilities there are certain areas like examination rooms and doctor chambers where camera coverage may breach privacy. These specific areas were termed as case-specific constraints.

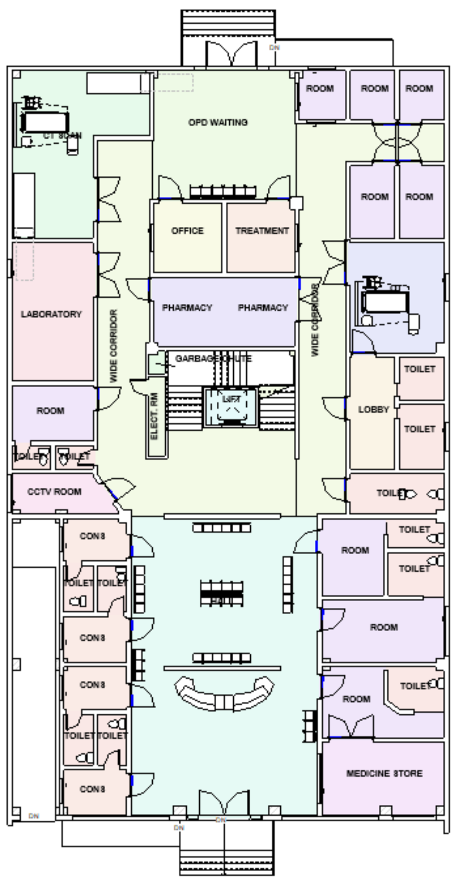

Figure 2 Plan view ground floor

Section 4.3: Experimental results:

For experimentation, user defined camera specifications like the FOV and range of cameras were taken as 88^{o} and 40ft, respectively.

Section 4.3.1: Manual camera placement

The camera layout, manually designed by the professional on the ground floor lobby, is shown in the Fig (3). Nine number of cameras were placed to provide a total of 66% coverage and 41% overlap region. High overlap percentage shows that the cameras are not placed efficiently, and inadequate percentage coverage implies presence of blind spots in the coverage area.

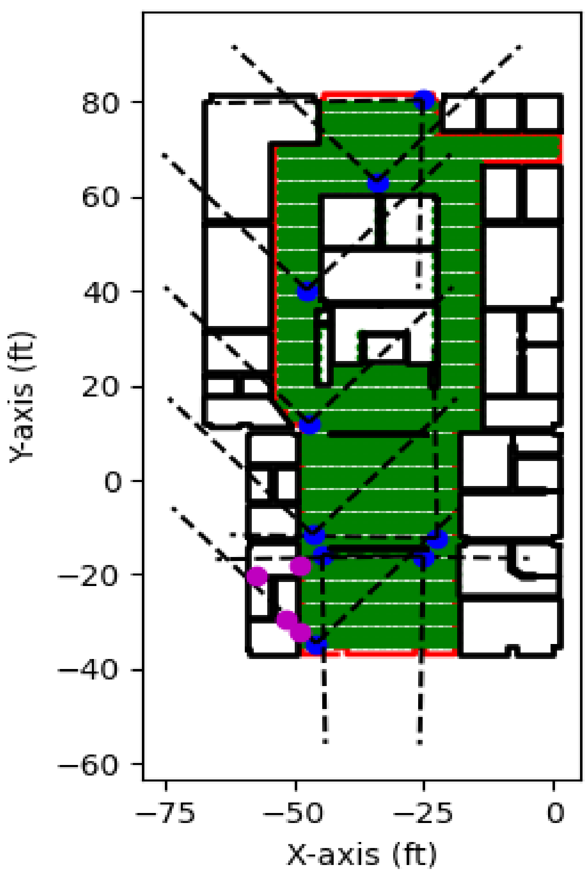

Figure 3 Manual Placement

Section 4.3.2: Optimized camera placement – Scenario A

Next, camera layout was designed using the optimization framework on the same region. For module A, Number of cameras were kept the same as used by the professional and percentage coverage was the subject of optimization while minimizing overlap. Rooms lying inside the lobby were considered as obstacles and main entrances and exits were defined as ROIs. Next, real coded genetic algorithm was implemented by setting following parameters population size as 500, number of generations as 100, Mutation rate as 0.3, 0.5 overlap penalty and 0.02 elitism rate.

The fitness of best individual increases throughout the generations, gradually converging to the most optimal solution. The fitness curve and the final results generated through the optimization process are shown in Fig (4 and 5). 82.18% coverage was achieved with 15.63% overlap region. The percentage coverage when compared to manual placement has increased by 24.94% and the overlap region is reduced by 61.34%.

Figure 4 Scenario A camera placement

Section 4.3.3:Scenario-A Fitness curve of GA

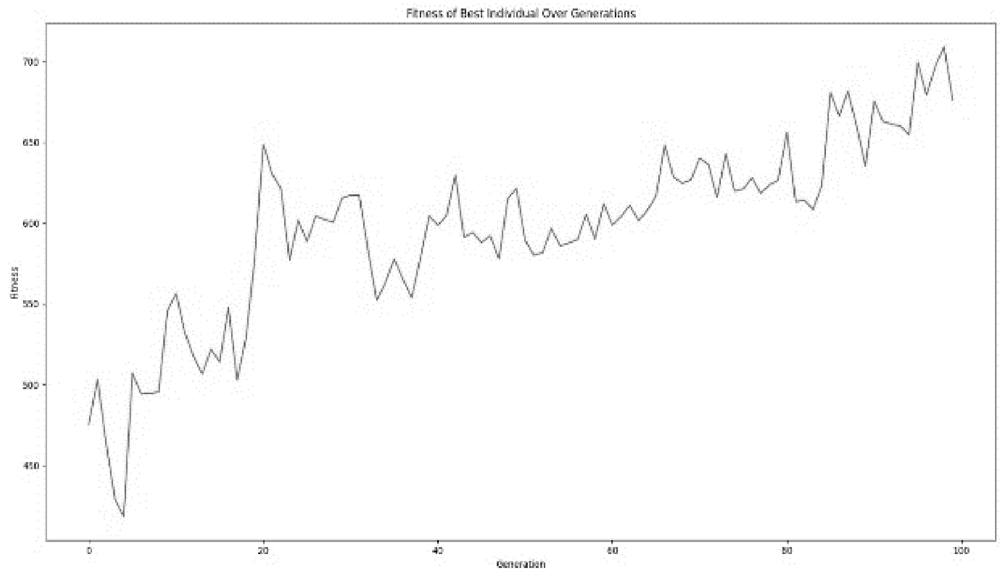

Figure 5 Scenario A fitness curve

This reduction in the overlap region indicates more efficient use of cameras and the increase in coverage area results in less blind spots when compared with manual camera placement.

Section 4.3.4: Optimized camera placement – Scenario B

Next, the number of cameras was taken as the subject of optimization keeping the rest of the parameters same, to get the desired coverage with minimum number of cameras. Results produced through this optimization process are shown in the Fig (6). Six number of cameras were used to provide 73.01% coverage with 1.1% overlap region. Fitness curve of this optimization process is shown in Fig (7)

The percentage coverage was increased by 10.6% while using three less number of cameras as compared to manually designed layout. All region of interests were also included in the coverage.

Figure 6 Scenario B camera placement

Section 4.3.5: Scenario-B Fitness curve of GA

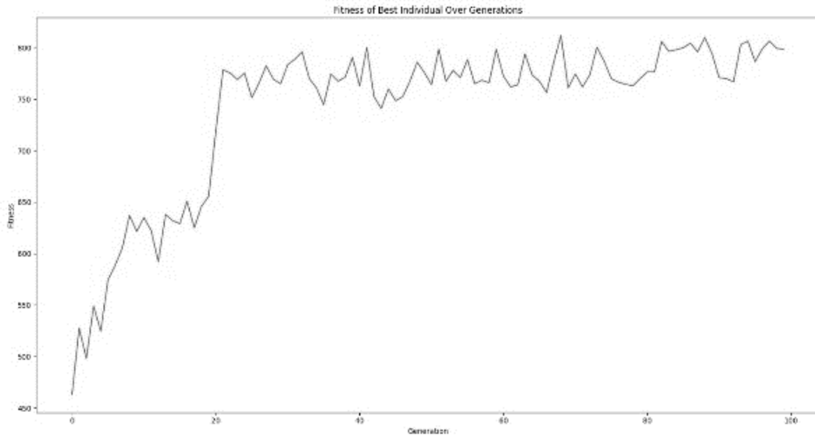

Figure 7 Scenario B fitness curve

This demonstrates that an optimized algorithm-based layout can achieve equivalent results as a manually designed layout but with reduced number of cameras.

Section 4.4: Implementing Results on BIM:

After the completion of optimization process, the resulting optimized cameras are integrated into Revit through the utilization of pyRevit plugin. Upon clicking the “import cameras” button within Revit, user is prompted to select the appropriate floor for camera placement. Once the floor has been chosen, the cameras are automatically positioned within the building’s plan view. Figure (8) illustrates the imported results of Scenario B

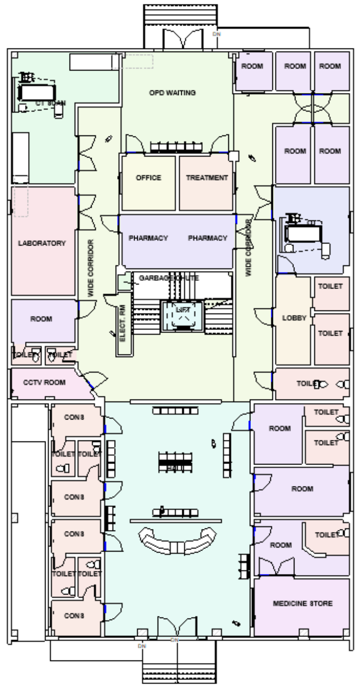

Figure 8 Camera placement in Revit

Figure 9 Camera View

Chapter 5:Conclusion:

A framework for optimized and automated camera placement is developed in this study. The proposed framework is able to generate the optimal position and orientation of cameras considering general and case specific constraints that specify user defined requirements. It aims to minimize the total number of cameras required and reduce any unnecessary overlap in their coverage areas. First the building coordinates including rooms and columns are extracted from BIM and coverage area and constraints are specified. Next, main program reads the required coordinates and prompts the user to input individual camera specifications, ROIs and percentage coverage required/number of cameras to be used. Then the camera network optimization problem is formed and solved using Genetic Algorithm. It starts with an initial population of potential solutions and then genetic operators are applied, enabling the algorithm to effectively explore through the search space gradually converging to the most optimal solution. The algorithm terminates when either the required percentage coverage is achieved, or the total number of generations are reached and returns the best solution. Lastly, the optimized results are imported back in BIM, cameras are automatically placed through a plugin in the position and orientation generated by the main program and their placement is visualized.

A case study was conducted on the ground floor lobby of River Garden Hospital to validate the performance of proposed optimization framework. Camera layout was designed and placed on this area by both, the proposed framework and manually by an industry professional and a comparative analysis was conducted to evaluate the results. It was found that the layout generated through the optimized process gives better results than a manually designed layout. When optimizing the coverage while keeping number of cameras same (Scenario A), coverage increased by 24.94% and had 61.34% less overlap area as compared to manually designed layout. In the second scenario number of cameras were also optimized and 10.6% more coverage than manual layout was achieved with three less number of cameras used. Also, the overlap was reduced by 97% which implies cameras being used to their full potential i.e., providing maximal coverage. In conclusion, the proposed method has shown better performance than the manual method including higher time efficiency, higher percentage coverage, reduced number of cameras and lower percentage of overlap.

Although the experiments have shown promising results, this study still has some limitations. First, the pan and tilt of cameras was not considered. Second, the element of height was not incorporated into the optimization process and all cameras installed were assumed to be at the same height. Third, the framework is resources intensive, and demands considerable hardware specifications to perform optimization with higher number of generations and a large population size. Further research can be done to make the framework less resource demanding and develop a user-friendly interface for convenient usage.

References

1. Amiri, M.R.S. and S. Rohani, *Automated Camera Placement using Hybrid Particle Swarm Optimization*. 2014.

2. Goldstein, E., *CCTV Helps Both Patients And Staff*. 2021.

3. Chen, H.-T., S.-W. Wu, and S.-H. Hsieh, *Visualization of CCTV coverage in public building space using BIM technology.* Visualization in Engineering, 2013. **1**(1): p. 5.

4. Qiu, Q., et al., *Scan planning for existing buildings without BIM based on user-defined data quality requirements and genetic algorithm.* Automation in Construction, 2021. **130**: p. 103841.

5. Albahri, A. and A. Hammad, *Simulation-Based Optimization of Surveillance Camera Types, Number, and Placement in Buildings Using BIM.* Journal of Computing in Civil Engineering, 2017. **31**.

6. Tang, K.-S., et al., *Genetic algorithms and their applications.* IEEE signal processing magazine, 1996. **13**(6): p. 22-37.

7. Chvátal, V., *A combinatorial theorem in plane geometry.* Journal of Combinatorial Theory, Series B, 1975. **18**(1): p. 39-41.

8. Horster, E. and R. Lienhart. *Approximating optimal visual sensor placement*. in *2006 IEEE International Conference on Multimedia and Expo*. 2006. IEEE.

9. Yabuta, K. and H. Kitazawa. *Optimum camera placement considering camera specification for security monitoring*. in *2008 IEEE International Symposium on Circuits and Systems (ISCAS)*. 2008. IEEE.

10. Xu, Y. and B. Lei. *Particle swarm optimization to improve the coverage of a camera network*. in *MIPPR 2009: Medical Imaging, Parallel Processing of Images, and Optimization Techniques*. 2009. SPIE.

11. Xu, Y., et al. *Three particle swarm algorithms to improve coverage of camera networks with mobile nodes*. in *2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA)*. 2010. IEEE.

12. Yao, Y., et al., *Can you see me now? Sensor positioning for automated and persistent surveillance.* IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2009. **40**(1): p. 101-115.

13. Han, Z., et al., *Camera planning for area surveillance: A new method for coverage inference and optimization using location-based service data.* Computers, environment and urban systems, 2019. **78**: p. 101396.

14. Zhang, C., V.S. Kalasapudi, and P. Tang, *Rapid data quality oriented laser scan planning for dynamic construction environments.* Advanced Engineering Informatics, 2016. **30**(2): p. 218-232.

15. Kang, F., J.-j. Li, and Q. Xu, *Virus coevolution partheno-genetic algorithms for optimal sensor placement.* Advanced Engineering Informatics, 2008. **22**(3): p. 362-370.

16. Žalik, B. and I. Kolingerova, *A cell-based point-in-polygon algorithm suitable for large sets of points.* Computers & Geosciences, 2001. **27**(10): p. 1135-1145.