1-1-2017

Towards Optimal Ptz Camera Scheduling

Sina Gholamnejad Davani
Wayne State University,

TOWARDS OPTIMAL PTZ CAMERA SCHEDULING

by

SINA G. DAVANI

THESIS

Submitted to the Graduate School

of Wayne State University,

Detroit, Michigan

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

2017

MAJOR: COMPUTER ENGINEERING

Approved by:

____

Advisor Date

ACKNOWLEDGMENTS

I would like to thank my adviser Dr. Nabil Sarhan from the bottom of my heart. His help and guidance and knowledge and experience were extremely helpful in my research. I also would like to thank my committee members Dr. Feng Lin and Dr. Hao Ying for their precious feedback.

DEDICATION

To all those that have been supportive of me in achieving my goals, especially my parents.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

Chapter 1: Introduction

Video surveillance systems have recently grown dramatically in popularity. The market share for this industry is expected to reach $75.64 billion in 2022 [1]. These systems can reduce the number of security personnel and required equipment, and at the same time provide the comfort and the peace of mind for the deployers of these systems.

Automated Video Surveillance (AVS) is the new trend in this industry which can be used in real time detection of threats and harmful activities. These systems take advantage of computer vision algorithms and have the benefit of independence from the human component of video surveillance systems. This characteristic makes them very interesting and desired as the human error element, which can result from tiredness in doing repetitive tasks, is eliminated.

Top-tier video surveillance systems use multiple Pan/Tilt/Zoom (PTZ) cameras which gives them full flexibility in covering the surveillance site throughout the view from different PTZ cameras, each covering a portion of the surveillance site. In actual deployments, PTZ cameras are currently controlled by humans or by predetermined scanning tours. However, automatic control of PTZ cameras is desirable. In most cases, the number of subjects traveling through the site is higher than the number of deployed PTZ cameras in the system. In these cases, the challenge is to determine a scheduling plan to dedicate each subject to a PTZ camera. Many studies have developed different solutions for this problem [2, 3, 4] (and references within), but they consider unrealistic assumptions and a small number of related elements in dealing with the issue. Additionally, these studies focus on capturing images with higher resolution and increasing the number of times a subject is captured. But the main goal in AVS systems should be maximizing the recognition probability of the subjects, present in the surveillance site.

In [5], a solution was introduced and implemented which sought to maximize the average subject recognition probability in AVS systems. This optimization was achieved by controlling the pan, tilt, and zoom of various utilized PTZ cameras by considering the attributes of the subjects in the surveillance area. Based on that study, the control of cameras depends on many elements, such as location of the subject, direction of movement, level of achieved satisfaction, distance to the camera, predicted departure time form the site, and the movements of cameras and their abilities and impediments. Camera movement is comprehensively characterized here as the time spent to set a specific pan, tilt, zoom, and focus setting. Their proposed solution considered 3D environments and it was not limited to 2D scenes. That study was the first camera scheduling solution which considered the influence of all these elements. The work was further improved by proposing and analyzing a clustering-based approach in [6], which can be used in conjunction with the camera scheduling schemes proposed in [5], enabling significant improvements in the subject recognition probability and the algorithm computation time. The clustering approach was applied to populate frames more efficiently, thereby enabling the system to focus on the areas that are populated with more subjects.

Three schemes were introduced in the proposed work in [5] to schedule the cameras in the surveillance site: *Brute-Force Grouping*, *Grid-Based Grouping*, and *Elevator-Based Planning*. These methods from this point forward will be referred to by BFG, GBG, and EBP abbreviations, respectively. In BFG a filtered frame list was created per each PTZ camera and then by selecting one frame from each camera, an arrangement of frames was created to schedule the PTZ camera movements. Aggregate recognition probability of all subjects was calculated for each possible selection of subset frames and the highest one was selected as the created scheduling plan. GBG was an improved version of algorithm in [2]. In [5], the authors enhanced the algorithm by considering cameras having different Fields-of-Regard, generating frames according to cameras' specifications and grouping in 3D area (Fields-of-Regard, FoR, is all the views that can be captured by a PTZ camera using different PTZ settings). EBP used the filtered frames and utilized them to create specific plans for every camera separately. The process incorporated the current PTZ settings of each camera and its speed in PTZ movements. During the recording period, EBP gave the ability to each camera to follow a different set of subjects, as opposed to BFG and GBG, which tracked the same group of subjects repeatedly. By considering the applied clustering approach, additional three scheduling schemes can be included in the original set of camera assignment policies. Namely, BFG-C, GBG-C, and EBP-C. The 'C' is added to differentiate from the base scheduling scheme and to specify the incorporated clustering approach.

The camera control solution operates into two alternating phases: pre-recording and recording. These two phases together create one cycle of operation in AVS systems. During the first phase, all the calculations and required observations are done to eventually create a plan of PTZ camera movements for the next stage. The next stage executes the calculated and built plan, resulted from the first stage, and thus each camera would follow the related plan created during the pre-recording time. If the pre-recording time is decreased to the minimum required value, the accuracy improves as the AVS system can concentrate more on the recording stage, increasing the chances of subjects to be captured during this stage.

In this thesis, we build on Studies [5] and [6], as they offer the best available solution in controlling PTZ cameras in automated surveillance systems. We enhance the recognition accuracy by developing a parallel control algorithm that reduces the time spent on the pre-recording phase. Our dynamic pre-recording approach can take advantage of this early completion of the pre-recording tasks and subsequently can reschedule the recording phase to an earlier point in time.

The parallel algorithm takes advantage of multi-core processing architectures and enables the system to scale well with the number of applied PTZ cameras. Subsequently, the parallel scheduling algorithm finishes the required tasks and computational duties, earlier than the sequential scheduling algorithms in each cycle of operation. This enables the system to scale with the overall monitored area of the deployed AVS solution, as increasing the area can be achieved by increasing the number of deployed PTZ cameras.

When the required computations are completed sooner, utilizing of the remaining time in the preparation stage for recording and tracking subjects, becomes necessary. As previously mentioned, to fully realize the potential of parallel algorithm, a dynamic pre-recording method in planning the PTZ camera movements is developed. By applying this new scheme, we enable the system to spend a larger portion of the total operational time on subject recording and tracking. This new approach uses a novel method to estimate the required pre-recording time in each operational cycle and then uses these values to predict and interpolate the future stages of the subjects in the surveillance site. After that, the recording stage is pushed back to an earlier schedule, when it is determined that all the required calculations and planning for each PTZ camera are complete. This method reschedules the recording stage sooner in comparison to the static pre-recording time approach.

This new approach, alongside the parallel algorithm in camera scheduling, greatly contributes to a more efficient surveillance system, producing the highest recognition probability for the subjects, traveling through the surveillance site. The focus is primarily on the face recognition algorithm and its accuracy is used as the metric guiding the camera controlling process as in [5] and [6].

We evaluate our proposed enhanced solution extensively through simulation. We consider more than 600 simulation settings, including various combinations of different subject arrival rates, pre-recording time, recording times, surveillance area sizes, cluster sizes, number of cameras, etc. We conduct more than 10, 000 simulation runs to analyze the effectiveness of proposed solution. Each simulation was executed multiple times to eradicate the effect of exceptional execution cases.

We also incorporate a new inclusive speed model, which considers the number of subjects, present in the unit area in the surveillance site. This value which is the subject density in the surveillance site, is a factor impacted by many elements, including social tendencies of the subjects. By adopting this inclusive speed model, we can achieve realistic evaluations of our proposed enhanced solution as the speeds of subjects in the surveillance site are determined based on a realistic speed model.

The main contributions of this thesis can be outlined as follows.

- We develop a parallel algorithm, which results in a much more efficient and faster operation in the preparation period (pre-recording period) before recording and tracking of the subjects traveling through the surveillance site.

- We introduce and implement a dynamic approach for determining the pre-recording time in the AVS system, thereby enabling the system to incorporate the available unused time in the pre-recording period and thus resulting in early scheduling of the recording and tracking phase, and consequently increasing the recognition probability for the subjects passing through the surveillance site.

- We apply a new and more realistic workload model incorporating many factors, including social characteristics, to achieve more realistic evaluations of our enhanced solution by updating and maintaining the speeds of the subjects in the surveillance site with regard to realistic inclusive factors.

- We evaluate the applied improvements through extensive simulations and present the results in a clear way to differentiate the effects of applying our enhancements.

The thesis is organized as follows. Chapter 2 discusses the background information and related work. This chapter describes face recognition, camera scheduling and assignment concepts, subject grouping policies, the formulation of the objective function, and the clustering approach. Chapter 3 presents the proposed solutions. First, an overview of the overall surveillance system is described and demonstrated at the beginning of the chapter. Then, the enhanced solution is described briefly to give a better understating of the operational scheme of the proposed enhancements. Subsequently, the used accuracy model is described briefly and later the enhanced PTZ camera scheduling scheme is explained. The chapter includes the presentation of the proposed parallel algorithm for PTZ camera control and the dynamic approach for determining the pre-recording time. At the end of chapter, the incorporation of the realistic pedestrian speed model is explained and the related factors in the model are assessed. Chapter 4 details the methodology for performance evaluation and the simulation environment. In this chapter, all the related details and environment settings for producing different figures are specified, allowing the repeatability of conducted experiments. Finally, Chapter 5 presents the main results, including informative figures and charts to effectively demonstrate the performance gains and benefits of our enhanced utilized solutions, by enabling an easy comparison of the proposed solution with prior solutions.

Chapter 2: Background Information and Related Work

In this chapter, we will discuss background information and related work on face recognition, camera scheduling and assignment, subject grouping methods, the formulation of the objective function, and the clustering scheme.

Main challenge is Face Recognition. Numerous face recognition algorithms and face recognition applications [7, 8] have been introduced. Many of the algorithms have been presented for specific issues, such as illumination, pose, pixel density, occlusion, and face expression. These solutions' requirements include, pre-enhancement by utilizing image processing algorithms, training on divers images for one specific subject, having more than one image sample on the dataset for the same subject, or extensive calculation complexity.

Face recognition applications [7, 9] have many restrictions. These applications incorporate a quality algorithm to examine the *faceness* of the image, or that is to say, the degree of which the face in the image is suitable for recognition.

The main elements influencing frame quality for face recognition are pose, resolution, zoom-distance noise, occlusion, illumination, and expression of the face [10, 11, 12, 13, 7, 9]. Subjects with pose angles more than ± 25 are deficiently recognized [11, 7, 9]. As for the captured resolution, a minimum inter-ocular of 60 to 120 pixels is the preferred resolution for face recognition algorithms, to operate sufficiently acceptable. Camera zooming can be incorporated to produce the desired pixel resolution to make the face recognition task easier, despite the fact that zooming on distance subjects can result in a blurring noise issue [12]. Partially occluded faces are also an issue and many face recognition algorithms try to resolve this problem [13].

2.1 Related Work on PTZ Camera Scheduling

The quantity of PTZ cameras in the surveillance systems is limited. One of the issues that need to be addressed is when the number of present subjects in the surveillance site is larger than the number of deployed PTZ cameras in the system.

PTZ cameras have attractive attributes, such as large *fields-of-regard* (FoR) and multiresolution views. FoR is the set of all conceivable *field-of-views* (FoVs) that can be created using different camera's settings, where FoV is the view observed by the camera utilizing one particular setting (pan, tilt, and zoom values). PTZ cameras in the surveillance systems are mostly manually controlled. In automatic use case scenarios, the PTZ cameras are controlled using a patrol program [14, 15, 16].

Study [17] demonstrated a typical master-slave architecture setting for controlling the PTZ cameras. Scene descriptive information is sent to a centralized proxy server by incorporating fixed wide-angle cameras. The proxy server examines the descriptive scene information and sends commands to control the PTZ cameras. Higher quality frames of interested subjects are captured and sent to the server. Each camera in this study was able to track only one subject at a time. A scheduling arrangement decided the next frame to be captured.

In [18], one subject was captured at each time, and after that the camera moved to the closest subject to the current one. In [19], applying a network packet scheduling scheme were studied. Study [20] used CPU scheduling in operating systems as a model for creating scheduling practice while cameras were treated as processors and subjects as jobs.

PTZ cameras used pre-calculated plans in Study [4], with each plan having a list of subjects to capture throughout time. A master-slave architecture was incorporated by which each PTZ camera received its detailed schedule from the server and then each camera executed the assigned scheduling independently. The study incorporated only a tracking application of people. It considered just a few components and studied just face detection.

Study [5] presented three schemes for camera scheduling: *Brute Force Grouping*, *Grid-Based Grouping*, and *Elevator-Based Planning*. These schemes are referred to as BFG, GBG, and EBP, respectively. In BFG a filtered frame list was created per each PTZ camera and then by selecting one frame from each camera, an arrangement of frames was created to schedule the PTZ camera movements. Aggregate recognition probability of all subjects was calculated for each possible selection of subset frames and the highest one was selected as the created scheduling plan. This probability is calculated by incorporating the recognition likelihood of each subject just once at its maximum value in the set, without considering the number of appearances. GBG is an adjustment of the algorithm in [2] with several improvements. The authors in [2] assumed that every camera seizes the same set of frames in the same scheme. Specifically, a frame can be seen by all PTZ cameras located at any position in the surveillance area, and at the same time it will result in the same recognition probability value. The entire solution was described in a 2D environment, avoiding complications of 3D environments. In Study [5], the algorithm was enhanced in four different approaches: (i) considering cameras having different FoRs, (ii) producing frames according to cameras' specifications and not site area dimensions, (iii) distributing the subjects in different groups by considering a 3D environment by incorporating the cameras' field of views and subjects' locations in the surveillance site, and (iv) considering overlapped frames in a 3D environment. In EBP, scheduling plans for each camera is created by purifying the filtered frames. The process incorporates the current state of each PTZ camera and its specifications. As opposed to other two methods, EBP enables each camera to view a different group of subjects throughout the time during one recording period, instead of tracking the same subjects during the whole period.

Subject grouping is another method utilized to handle the scalability issue by grouping or putting multiple subjects in one frame and assigning that frame with all the included subjects to one PTZ camera. Paper [3] modeled the surveillance architecture as a bipartite graph. The subject-to-camera allocation was determined by incorporating the maximum matching algorithm. This work did not represent any clearly defined distinction between frames and groups in their proposed approach.

Lattice-Based Grouping algorithm used in paper [2] and incorporated a master-slave architecture by employing a wide-angle camera to gather and send scene descriptive data to the processing unit which examined the information and produced a set of frames that amounts to the highest "Objective Satisfaction". PTZ cameras then captured the assigned frames in their related plans with a high resolution. The paper considered that all PTZ cameras employed in the surveillance site have completely the same FoR (located at the same place). This assumption oversimplifies the issue and severely reduces the search domain for candidate frames to be used in the plan building process. Also, the study didn't consider subject's standing pose or occlusion, which can be an important factor in the face recognition tasks.

Additionally, the "subject satisfaction" was computed in a linearly-additive model, avoiding the probabilistic nature of the recognition scheme.

In [5], authors have developed a solution for optimizing the overall subject recognition probability by determining the optimal PTZ movement values for each utilized PTZ cameras. This solution considered the recognition probability, dealt with PTZ cameras incorporating different FoRs, produced frames by considering the capabilities and limitations of cameras and examined the overlap among frames in a 3D environment.

The developed solution took into account the movements of PTZ cameras, working on an elevator-based fashion. In a later work, they have also proposed and analyzed a clustering-based approach, which can be used in conjunction with the two aforementioned schemes, to make significant improvements in both the overall subject recognition probability and the algorithm computation time.

To populate frames effectively, Study [6] incorporated clustering as a pre-step for generating potential frames. The fundamental preferred standpoint of the clustering process was empowering the framework to concentrate on the regions that are populated with more subjects. In that study, subjects were gathered in light of many characteristics, which was interpreted eventually into distance. The following steps were taken to distribute subjects into different clusters.

1. Allocate nodes to clusters by searching and finding the nearest cluster to each node. The distance is calculated based on many attributes which can all be transformed into a single scalar value. Clusters are formed based on center, direction of motion, and size.

2. Determine new cluster center and repeat Step 1.

A form of the *dendrogram hierarchical clustering* technique [21] was used, by which each subject was considered as a cluster center initially. To examine if a subject fit in to a cluster, three parameters were explored and examined whether they were smaller than previously determined threshold values. These thresholds were the maximum permitted distance between the cluster center and the subject, the widest angle between the center of cluster's direction of motion and the subject's movement direction, and the maximum perpendicular-distance between the subject and the cluster's course of movement.

Subsequently, after determining the clusters, they have populated frames by considering each camera's related list of clusters. Each frame for a camera is produced by finding the required PTZ settings for that camera in order to effectively capture that cluster of subjects.

After producing the frames by incorporating the clustering scheme, they apply the two camera scheduling algorithms in [5], particularly GBG and EBP. Furthermore, they introduced and applied a new improvement, called *reachability enhancement*, to decrease the search domain. In particular, they enhanced the filtration by removing all the frames that were unreachable, owing to a long time of computation and required change in PTZ settings for deployed cameras.

Chapter 3: Proposed Enhancements Utilizing Dynamic, Parallel methods with Realistic Speed Models

3.1 System Overview

In this thesis, we utilized the system developed in [5] and [6]. In our proposed AVS framework enhancement, wide-angle cameras, PTZ cameras, and a processing proxy station are utilized. An oversimplified view of the system is demonstrated in Figure 3.1.

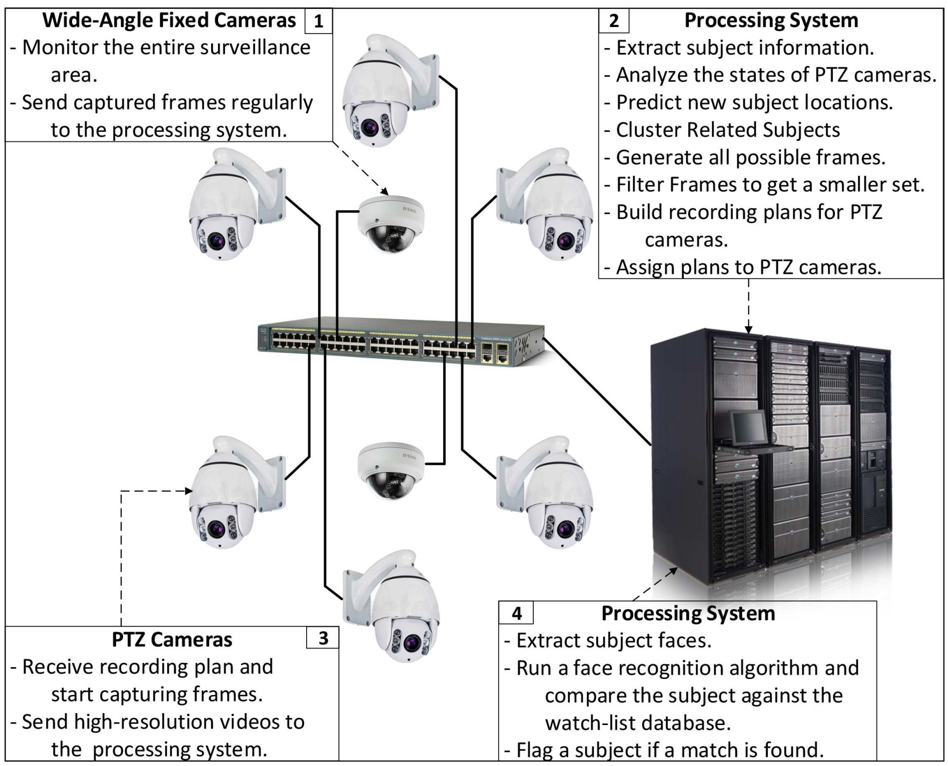


Figure 3.1: Illustration of the Overall System/Solution

Wide-angle cameras are employed to watch the site and send the video data to a processing system, which examines the scenes and subsequently controls the PTZ cameras. The PTZ cameras catch higher quality frames of the subjects.

The processing architecture executes the following undertakings. (1) It examines the pictures from the wide-angle cameras. Then, it uses this data to extract the location of the subjects and their specifications like speed and the angle of movement. This task uses the broadly inquired pedestrian detection algorithms [22] for discovering the different properties of subjects. (2) It forecasts the location of the subjects when the cameras will begin recording, which in static operation mode is clearly specified as oppose to the dynamic operation mode which uses an estimation to determine the start of recording stage. (3) It runs a PTZ camera scheduling solution which results in managing the PTZ camera movements in the surveillance system. This task may incorporate creating sets of conceivable frames as indicated by the states of the PTZ cameras and their capacities and constraints and can exploit simultaneous execution on the order of number of cameras in the surveillance site. (4) It receives high quality frames and then computer vision algorithms are executed on these frames. As an example, face recognition could be selected as the desired algorithm.

The system works in a cyclic mode and each cycle can be divided in to two alternative periods. *pre-recording period* (i.e. preparation) and *recording* [2]. The pre-recording period of T_p seconds is required for the processing system to predict the subject attributes and run the camera control algorithm. In the proposed dynamic operation mode, T_R will be used as an upper limit on the pre-recording time value. During the recording period of T_R seconds, high quality pictures will be captured by PTZ cameras and will be transferred back to the proxy station, which will consider them for additional processing. This period includes PTZ camera subject tracking operation. The resulting scheduling plan for each specific PTZ camera will be assigned in the first period. In the second period, the framework utilizes a *Watch List* which incorporates a picture database of subjects that are regarded dangerous.

3.2 Overview of the Overall Solution

Scheduling the PTZ cameras and assigning subjects to PTZ cameras in realistic environments are the main addressed problems by our proposed enhanced solution.

The main objective for this enhanced solution is to improve upon the best available solution and to enhance the average subject recognition probability. This goal is achieved by efficiently controlling the PTZ camera settings, considering the influencing element of subject characteristics in the surveillance site. This control of cameras in the best current available solution [6] considers many characteristics of the subjects, including location, speed, distance, the current satisfaction level, the predicted departure time from the surveillance area, and the deployed PTZ camera specifications like panning, tilting, and the zooming speeds.

We utilize and adopt the solution in [6, 5] and propose two main enhancements. The first enhancement is a parallel application which results in a much more efficient and faster operation in the preparation period (pre-recording period) before recording and tracking of the subjects who are traveling through the surveillance site. The second enhancement is the design and implementation of a dynamic approach for pre-recording time in AVS operations which enables the system to incorporate the available unused time in the pre-recording period, resulting in early scheduling of the recording and tracking phase, which consequently increases the recognition probability for the subjects passing through the surveillance site.

As demonstrated in Figure 3.2, the improved solution consists of the following main phases.

3.2.1 Clustering of Subjects

As in [6], we utilize clustering as a pre-step for generating frames. This enables the system to focus on the areas that are populated with more subjects. Subjects are placed into different groups by considering many attributes and each cluster is defined by three characteristics of cluster center, cluster direction, and cluster size.

The cluster center to subject distance, the angle between subject and the cluster direction of motion, and perpendicular-distance between the subject and the cluster direction of motion is examined in the mentioned study to determine if a subject is owned by a cluster.

3.2.2 Generation of All Possible Camera Frames

In this stage, the handling framework uses the data provided by wide-angle cameras to identify all visible subjects and calculates their characteristics, like location, speed, and course of movement. The characteristics can be resolved by utilizing one of the widely explored pedestrian detection algorithms [22].

The location of the subjects is calculated and estimated by the proxy station for the starting time of

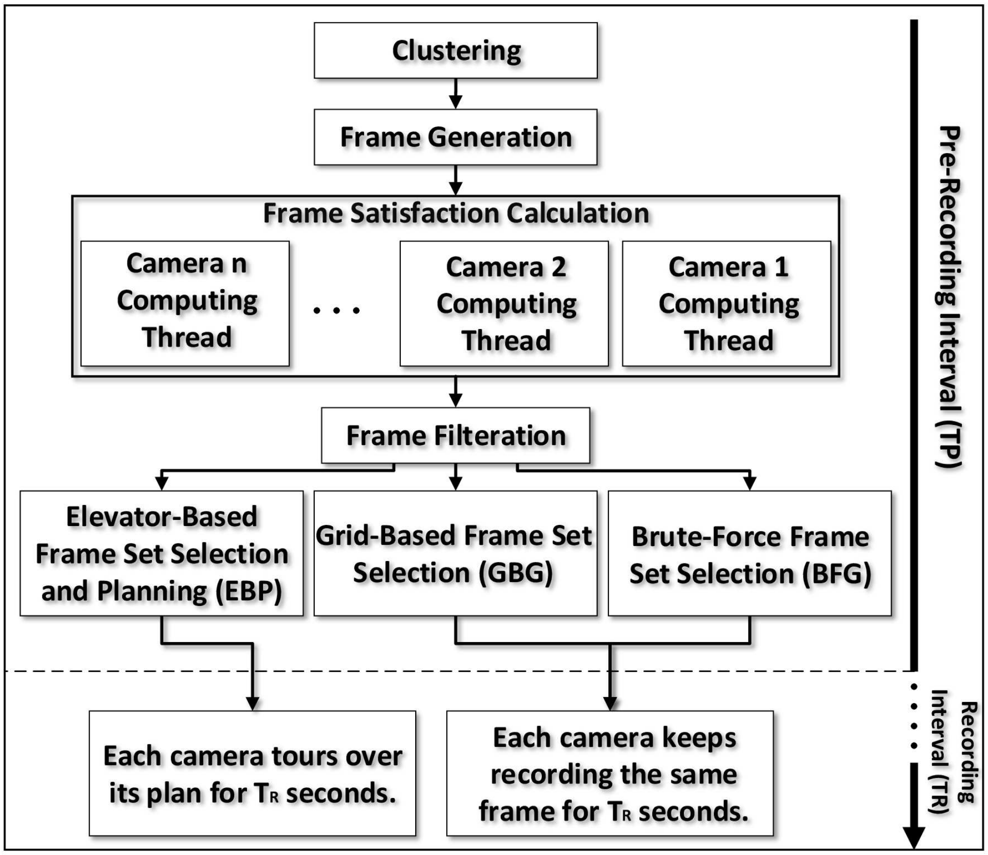


Figure 3.2: Illustration of the Proposed Solution [Ideally Recording Interval Should Be Longer Than or Equal to Pre-recording]

the recording period when the PTZ cameras will start capturing the assigned frames. This future point in time is determined in the static operation mode, but in the dynamic mode should be estimated as there is no procedure to determine the value accurately until after the camera planning process is over. By considering different pan, tilt, and zoom values for PTZ cameras and the future location of the subjects, all candidate frames can be produced for each camera. This last step utilizes the capabilities and limitations of the PTZ cameras and can be implemented by taking advantage of simultaneous execution on the order of number of PTZ cameras which makes the system greatly scalable. It should be noted that each PTZ camera is capable of recording a different frame using a different PTZ setting. The frame here is the view as seen by the camera at a particular PTZ setting. By exploring alternate settings, a set of frames covering the entire FoR will be produced. Joining all achievable frames, gathered from every camera, generates the frame domain which will be examined in later stages. Rotation and translation matrices are utilized to relate the 3D coordinates to each camera coordinates [23, 24].

3.2.3 Filtration of the Set of Possible Frames

This step filters the frames produced in previous stage by removing the frames that do not capture any subject in the surveillance area and by choosing only the finest camera frame from the sets of frames that catch the very same arrangement of subjects. The selected frame is the one that produces the highest combined recognition likelihood.

3.2.4 Assigning of Frames and Camera Scheduling

In this phase, the processing system executes the camera scheduling and frame assignment duties. The proposed enhanced schemes for this task which includes the dynamic pre-recording approach are explained later in this chapter.

3.2.5 Recording of Frames and Tracking of Subjects

In this stage, the PTZ cameras incorporate the settings of camera planning and scheduling to capture the previously considered subjects and begin recording and tracking these subjects, as it is described in [6].

3.3 Used Accuracy Model

Numerous face recognition algorithms [8, 7, 9] are created to handle complexities like the pixel density and occlusion.

The inter-ocular (between eye pupils) distance should be at least 60 to 120 pixels for face recognition algorithms for a proper performance [25, 26]. To keep this pixel density for moving subjects, the camera's zoom has to be modified with respect to the subject distance.

The time a PTZ camera spends covering a specific frame needs to be selected cautiously. If the camera spends more time on one frame, the included subjects will have better chances for recognition and at the same time this may decrease the chance of other subjects to be sufficiently captured and recognized.

In [5], an objective function for the recognition probability is proposed. The recognition probability $\gamma(s, f)$ of subject s placed in camera frame f, based on this study is formulated as follows:

$$\gamma(s, f) = W(s) \times P_{pose}(\theta_{sf}) \times P_{zoom}(z_{sf}) \times u_{coverage}(s, f) \times u_{occultation}(s, f), \tag{3.1}$$

Table 3.1 describes different constituting functions of Equation (3.1).

Table 3.1: Description for Different Factors in Equation (3.1)

Factor	**Full Name**	**Description**
$W(s)$	Weight	Captures the subject's estimated departure time from the surveillance site and the current level of satisfaction for the subject under investigation
$P_{pose}(\theta_{sf})$	Probability Related to Pose	Relates the recognition likelihood to pose θ_{sf}
$P_{zoom}(z_{sf})$	Probability Related to Zoom Level	Relates the recognition likelihood to zoom level z_{sf}
$u_{coverage}(s, f)$	Coverage Status	Relates to the coverage status; the output value for this function is 1 when subject s is included in frame f
$u_{occultation}(s, f)$	Occlusion Status	Relates to the occlusion; the output value for this function is 1 when subject s is not occluded or covered by other subjects when looking through frame f

The use of predicted time for the subject to leave the site is taken from the weighting policies in [19, 20, 2]. After a subject is considered adequately recognized, the weight factor turns to zero and further activity may be compulsory whether the subject is determined to be in the Watch List or Trusted List. If the subject belongs to the Trusted List, the system may not require to dedicate any resources on the subject further. If it belongs to the Watch List, then the system should inform the administrators and arrange the capability to track the subject. The subject may no longer be examined for recognition purposes but it turns in to a high priority for other activities, like tracking.

Parameter θ_{sf} is the angle created by a line, crossing the face center of subject s and the line creating the optical axis of camera f. Zooming-in, particularly for far locations brings a blurring effect noise, resulting in a reduced recognition probability. z_{sf} is the zoom level used by camera f when it is looking at subject s, and the minimum 60 inter-ocular pixels requirement is met. Pose variation severely influences the recognition probability. Subjects with poses more than 25°are barely detected and recognized using normal recognition algorithms. To address this issue, different pose related face recognition algorithms were introduced. These algorithms require more computational power and longer calculation time to function and they perform properly, exclusively in controlled settings.

A subject can be placed outside of a camera's FoV. It is also possible for a subject to be covered by other subjects when looking through a certain camera' FoV. Functions $u_{coverage}(s, f)$ and $u_{occultation}(s, f)$ are used to assess the aforementioned statuses of a subject.

As in [5], the enhanced system considers the dependency between consecutive frames of the camera when it is calculating the aggregate recognition probability. When the frames are captured by a single camera with the same PTZ setting values, it is possible for these frames to be dependent if they are grabbed in a very narrow time window. The extra captured frames by this camera may not increase the recognition probability of the subject, as the subject's state may not have changed during this short period of time, creating almost two or more identical pictures of the subject. The parameter *dependency period* (T_{dep}) controls the recognition probability calculation in process, declaring all frames seized in this time window by a single camera utilizing a single PTZ status, dependent. In this set of dependent frames, one with highest recognition probability will be considered in calculating the overall related subject recognition probability.

3.4 Utilized Camera Assignment Approaches

In this thesis, we adopt the scheduling algorithms proposed in [5] and the clustering method proposed in [6], namely, BFG, GBG, EBP, BFG-C, GBG-C and EBP-C. A brief description of these schemes was given in Section 2.1.

EBP operates as follows.

- Seed frames are generated for each PTZ camera.
  - Each PTZ camera frame list is ordered by considering the recognition probability of the frame.
  - A specific number of highest frames in the ordered list for each camera is selected to decrease the computational complexity of next stages.

  - For each camera, out of the selected highest frames, a subset of *seed* frames is selected, such that this subset accomplishes the best total or aggregate recognition probability of all subjects.
  - This subset of frames is recorded and kept in *planSeedSet*.

- The plan list is created for each camera by using the selected seed frames from *planSeedSet*.
  - Each PTZ camera's *planSeedSet* is investigated and the frames are sorted based on the camera zooming value.
  - An ordered subset of frames, possibly with repeated elements of seed frames, is generated.
  - The ordered list is used to scan the surveillance site by moving through selected frames in an elevator-like fashion which is by moving from one frame to the next, until reaching one end, and afterward goes in reverse, and rehash the cycle, as long as the recording period contains unused, unscheduled time.

The aggregate recognition probability of all subjects is calculated by incorporating the recognition probability of each subject, only when it has the highest recognition probability value in the current frame in comparison to the other seed frames. This scheme does not consider the number of appearances in other selected seed frames which is a realistic decision. In Study [5] particularly, the aggregate recognition probability is described as:

$$\sum_{s}^{SCnt} \sum_{f}^{SeedCnt} (\bar{\gamma}(s, f) \times u(s)_{max}), \tag{3.2}$$

where $\bar{\gamma}(s, f)$ is simplified form of the $\gamma(s, f)$ described in Equation 3.1, created by removing the weight factor, $seedCnt$ is the number of selected seed frames, $SCnt$ is the number of subjects present in the selected seed frames, and $u(s)_{max}$ for subject s is 1 when the subject has the highest recognition probability in the frame, currently being assessed. Otherwise this value is 0. In other words, the subject is only considered one time in calculating the aggregate recognition probability for all the selected seed frames, which is logically true.

The operations of described stages in EBP scheme, occur in the related EBP scheduling section of Figure 3.2.

As detailed in Section 3.6, we proposed in this thesis an approach that considers a dynamic pre-recording time. The starting time for recording phase will be estimated based on the previous required pre-recording times, consumed in each preparation and tracking cycle.

The movement time which is spent on changing a camera's PTZ setting to cover a new frame is considered in EBP scheme, which dictates a new PTZ setting for that camera. This time can be described as the time spent on incorporating a particular pan, tilt, zoom, and focus setting for the PTZ camera, and can be defined by $T_{move} = max(Pan, Tilt, or Zoom Movement Time) + Focus Time$. The PTZ camera can move simultaneously in P, T and Z directions and subsequently, the measured maximum value for pan, tilt and zooming time, can be considered as the PTZ movement time. The frame recording time t_e, which is spent on capturing a candidate frame from seed frames, is also utilized in the scheduling scheme. The plan for each specific camera is assigned separately. The algorithm determines the best seed frame in the ordered list to begin with, by considering different plans created from various starting seed frames and selecting the plan with the maximum recognition probability. In the enhanced algorithm, this decision is depending on the starting time for the recording phase which will be estimated using a novel method. Figure 3.3 demonstrates by an example the scheme of creating a plan by incorporating the elevator-based policy.

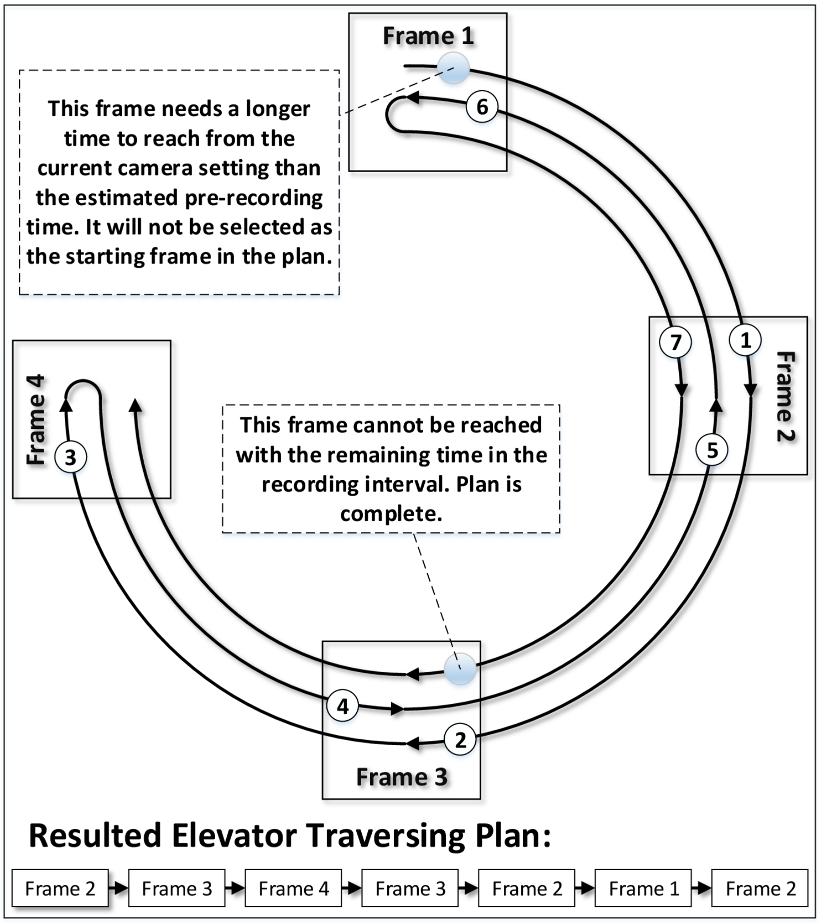


Figure 3.3: Creating a Plan by Incorporating a Seed Frame Set

A seed frame set with four elements is demonstrated in Figure 3.3. Frame 1 is avoided because there is not enough time left in pre-recording window to reach this frame, considering the present PTZ camera settings. The system continues adding seed frames to the plan while the recording phase T_R contains unused, unscheduled time. It scans the frames from top to bottom and from bottom to top or clock wise and counter clockwise, until the process finishes at frame 3 as it is not possible to include that frame in the plan because the recording time will exceed the assigned permitted value. This scheme is reiterated for different starting frames and all generated plans are compared. Subsequently, the plan with the highest recognition probability is selected and the PTZ camera will adopt this plan to execute during the recording period.

3.5 Proposed Parallel Algorithm for PTZ Camera Control

We have implemented a parallel algorithm for the PTZ camera control problem. In order to make this possible, the AVS system has to be divided to different operational units, including clustering, frame generation and satisfaction calculation, frame filtration and frame set selection and planning. All the units were timed multiple times with different system characteristics and all the units were initially implemented as parallel units in order to determine if the synchronization and communication costs of the operational units will be compensated by the gain in performance advantages of simultaneous threads of execution. In order to measure the execution time for these different units, the application was equipped with average time measurement functionality for different units and the average time was reported at the end of each simulation run.

In the implementation process, in order to achieve the highest possible performance efficiency, we have to make sure that the synchronization and the communication between different parallel units of execution are kept as an absolute minimum. As a result, hundreds of shared global variables used in frame satisfaction calculation unit were carefully examined to assess the possibility of moving them in to individual units of parallel execution as locally created variables.

By doing so, we can be assured that the contention between different parallel units, would be kept minimal which is an absolute necessity in parallel applications, as having shared data structures between multiple units of execution, dictates a mutual exclusion access to that data structure, nullifying all the performance advantages gained by having parallel units of execution.

One of the shared global data structures which had to be kept as a shared variable in parallel units, was an array for keeping the list of present subjects in the surveillance area. This list had to be accessed by different units of parallel execution and should be updated/edited constantly during the simulation runs, to make the solution process applicable. Consequently, controlled access was necessary, but having one mutual execution element to give control accessed to this array would severely hurts the performance gain as the array is considerably lengthy and being accessed numerous times during the simulation runs by different units of parallel execution.

This translates in to severe contention between parallel units to gain access to this array, causing a very narrow sequential bottleneck for the parallel application. One solution to this problem is to consider individual units of mutual exclusion for each element inside the array. Although this will minimize the contention problem between different parallel units, it will not be considered an efficient solution as the memory cost of having a one-by-one map array of mutual exclusion elements is high and not scalable, as it grows significantly with the same speed as the subjects list.

The best solution for the aforementioned contention issue, is to consider a fixed array of mutual exclusion elements and dedicate a set of subjects to each element. In order to minimize the chance of accessing one element by two different units of parallel execution simultaneously, we should distribute and scatter related subjects of a specific mutual exclusion element, as far and as distant from each other as possible in the shared mentioned array. The following sequence of operations would implement this perfectly optimized solution:

- $MutualExclusionElementsForSubjectsList[SubjectID \% ConstantSize].$ $GainControlledAccess();$
- $Access\ and\ Edit\ the\ Desired\ subject\ in\ The\ Subject\ List;$
- $MutualExclusionElementsForSubjectsList[SubjectID \% ConstantSize].$ $ReleaseControlledAccess();$

By incorporating the modulo operator on the $subjectID$, we can evenly distribute the subjects which are dedicated to one specific mutual exclusion element, throughout the subjects list array and at the same time, decreasing the number of required mutual exclusion units and keeping it at a controlled constant value, decreasing the inefficient memory requirement of having one mutual exclusion element for each subject.

In this proposed parallel solution, we can have up to $ConstantSize$ simultaneous parallel operatives, working on the same array at the same time without causing any illegal access to the data structure which puts the integrity of the solution in compromise.

By examining the experimental results in numerous settings and situations, it is very rare for a group of subjects related to one mutual exclusion element to be accessed simultaneously by multiple units of parallel execution which further validates our choice of design to enforce integrity in accessing the shared subject list data structure.

The granularity of parallel implementation was selected as the number of cameras in the AVS system which results in a perfect scalability of the system with the increment of number of cameras. It was determined that by implementing only the Frame satisfaction calculation unit as a parallel algorithm, as it is demonstrated in Figure 3.2, the performance advantage will be gained and the synchronization and communication costs will be smaller than the performance gain in simultaneous execution. Figure 3.4 demonstrates the simplified algorithm for parallel frame satisfaction calculation.

In subsection 5.4, we will illustrate the computation time for different units of the AVS system to clarify our policy in parallel implementation.

3.6 Dynamic Approach for Determining the Pre-recording Time

In order to take further advantage of our parallel PTZ camera control solution, we introduce a new dynamic approach to handle the required operations which is executing during the pre-recording time. During this period if all the cameras are done with their related computations and have determined that the movement time to the frame in the plan that will produce the highest recognition probability, in addition to the focus time will not violate the pre-recording time window, the highest required value amongst different cameras can be selected and based on this selected value the beginning of the recording phase can be adjusted, saving precious unused time and increasing efficiency and finally resulting in producing a higher recognition probability for the subjects, present in the surveillance site.

```
//Input: List of all the subjects in the surveillance site
//Output: list of frames for each camera with calculated satisfaction values
if (No more active subjects are left in site)
    return;
initialize the global Barrier variable to be used in
  FrameSatisfactionCalculationFunction thread;
totalFrameCountWithZeros = 0;
initialize(AggregateframeList);
initialize(occludedSubjectsArray)
initialize(FrameListForEachCamera);
initialize(FrameCountForEachCamera);
for(each camera c)
{
  create a thread using FrameSatisfactionCalculationFunction, using
   the initialized data structures as arguments;
}
for (each camera c)
{
  wait on the related thread to be finished;
  totalFrameCount += FrameCountForEachCamera[c];
  AggregateframeList.pushBack(frameListForEachCamera[c]);
  clear frameListForEachCamera[c];
}
clear FrameCountForEachCamera;
release the created threads;
```

Figure 3.4: A Simplified Algorithm for Parallel Frame Satisfaction Calculation

In Figure 3.5, a simplified illustration of the dynamic pre-recording method is demonstrated. Using this method, we can salvage the unused portion of the pre-recording phase and use it to conduct the recording and tracking activities.

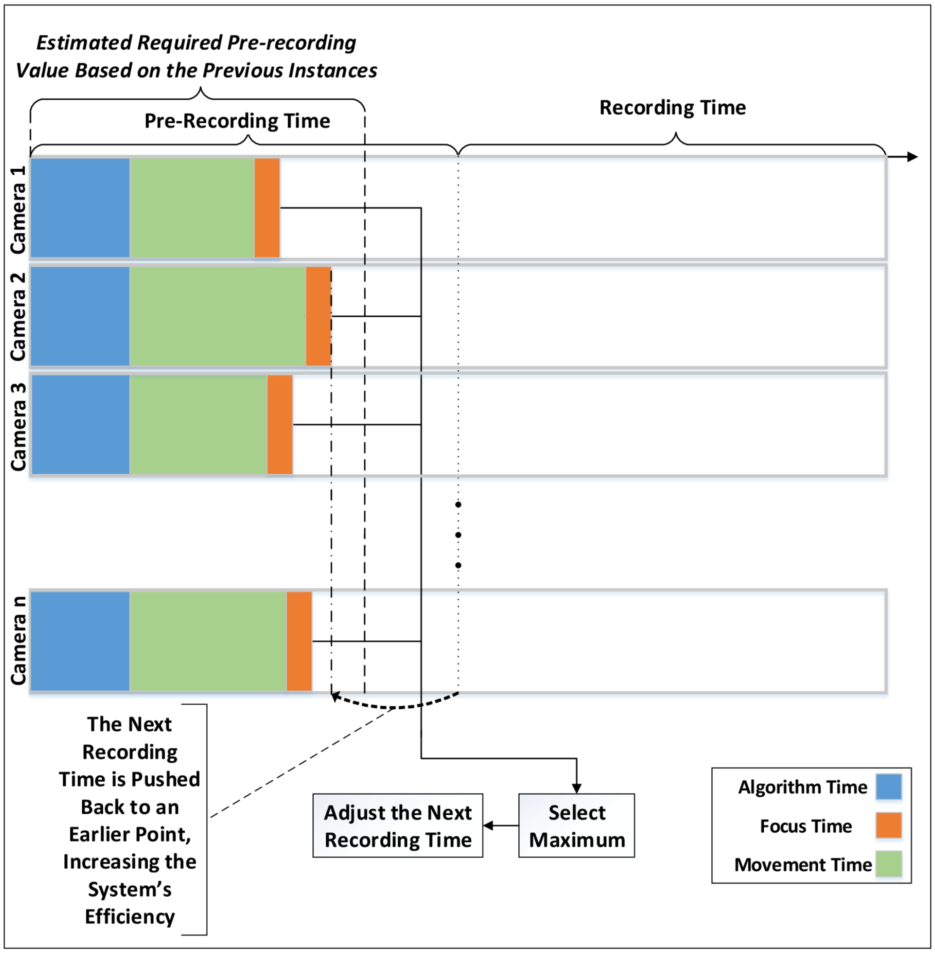


Figure 3.5: Illustration of the Dynamic Pre-Recording Method

When each camera tries to find the best frame in the seed array which would result in the highest recognition probability, it would consider a threshold value which should not be violated when all the required operations, including movements and computations, are complete. This value in the static approach is the pre-recording time which dictates that the starting point of the scheduled recording phase will not be missed. But in our new dynamic approach, this threshold value is selected base on the values from previous maximum required pre-recording times. By using the following equation, we determine this value.

$$Threshold = mean(MaximumRequiredPreRecordingTime, C \times previousThreshold) \quad (3.3)$$

The starting value of $threshold$ is the pre-recording time and C is a constant. By using a C value greater than 1, we can ensure that the solution will have the ability to increase the $Threshold$ value, according to the workload changes in the surveillance site which will be desperately needed if there is a sudden increase in the number of present subjects in the surveillance site. This sudden increment results in a spike in the amount of required computations in order to build a recognition probability maximizer plan for each camera.

Figure 3.6 demonstrates our enhanced optimized solution utilizing the dynamic pre-recording time. Lines 1 to 7 initialize different data structures, used in the later plan building process and sort the frames in each camera list based on the zoom value. Line 8, checks to see if the selected seed node in the considered camera seed array is eligible to be selected as the first frame for starting the plan based upon. Lines 10 to 14 add the first seed frame to the plan list and update the related values based on this first seed frame selection. lines 15 to 25 add consecutive seed frames to the plan from the sorted seed array, as long as the recording time allows. Lines 16 to 19 implement the back and forth movement scheme among the plan building frames. Lines 20 to 23 add the currently considered seed frame to the plan, if there is still time left in the recording period. Lines 27 to 31 check to see if the currently selected starting seed frame has produced the highest recognition probability and if so, it selects the plan as the current best plan for the currently being investigated camera and records the system lead time spent to create this plan. Lines 33 to 35 will make sure that the maximum required time for pre-recording period is calculated, which covers the operational time windows of all different cameras. Line 36 adjusts the starting time of the incoming recording period. The $nextTrackTime$ is adjusted by using the maximum required pre-recording time value which in turn is calculated by taking in to account, the current required pre-recording time for a specific camera (T_{SLR}). The threshold value which should not be violated in finding the best frame to start the plan building process upon on, is adjusted in line 37 by using the previous threshold value and the maximum required pre-recording time determined during previous operations (T_{MSLR}). Line 8 and lines 30 to 38 included the new main changes which will enable the system to dynamically adjust the pre-recording period and reschedule the next recording phase to utilize the unused, remaining portion of the pre-recording time.

By considering a dynamic pre-recording value, the prediction task which is a vital operation in determining the candidate frames to build a recording plan for each camera, gets complicated as the beginning of the pre-recording time is unknown, in contrast to the static method. In order to alleviate this complexity, we incorporate the previous maximum required pre-recording time in each period to convey the prediction task.

One important fact here which should be mentioned is the existence of the pre-recording time value when the dynamic pre-recording method is utilized. The question to ask here is, why there is a need for the pre-recording time value when the dynamic method automatically determines the perfect best value and continues with the recording phase.

To answer this question, let us examine the benefits of having this value to be determined and used in the surveillance system scheduling solution. The first benefit can be described as having ability to force a strict deadline on the operations, required to be executed during the pre-recording period. Let us imagine an unusual severe increase on the number of subjects entering the surveillance site. By using a threshold value as the pre-recording time, we can ensure that the system will not spend majority of its time portions in preparation for the recording phase and missing many opportunities to record some crucial and helpful scenes from the surveillance site.

On the other hand, if an administrator of a surveillance system decides to have the solution implemented on a system configured as a full automatic dynamic pre-recording mode, he/she still can achieve this goal by simply selecting a very large value for the pre-recording time which strongly has the same effect as using an implementation, using full automatic pre-recording concept.

3.7 Incorporating a Realistic Pedestrian Speed Model

In contrast with [5] and [6], we use a realistic inclusive speed model to update the subjects speed value, who are present in the surveillance site. This new enhancement, considers many important factors, including subject density. Over the last fifty years, the popular Bureau of Public Roads (BPR) model [27] has been extensively used by policy makers and traffic researchers to determine travel times and consequently speed, on road networks. This model can also be adjusted to the following form, in predicting pedestrian travel time and speed on different walking sites:

$$t(k) = T_0 + A \times (k/k_j)^s, \tag{3.4}$$

where $t(k)$ = moving time (sec) at density k, T_0 = free-flow moving time (sec), k = pedestrian flow (peds/m/sec) or density (peds/m^2), A and s = constants to be estimated in the model fitting procedure, k_j = the capacity of the pedestrian site (peds/m/sec or peds/m^2), and k/k_j = pedestrian density to capacity ratio.

This model, as well as others which were developed based on it, do not explicitly take into account the local pedestrian body dimensions (e.g. the lateral spacing for movements). Additionally, these models have limited ability to supply a platform for sensitivity analysis. The queuing theory based model introduced in [28], beside the fact which is an abstract of reality, conquer these problems and can formulate the sophisticated and stochastic pedestrian movements and travel time in a thorough manner. The authors in this study, introduced the following pedestrian speed model:

$$v = \frac{v_f}{[1 + \frac{(k/k_j)^s}{s(1-k/sk_j)(k/k_j)^s + s(s!)(1-k/sk_j)^2 \sum_{n=0}^{s-1} \frac{(k/k_j)^s}{n!}}]}, \tag{3.5}$$

where s = the number of pedestrians that can be accommodated in the minimal length of the facility =

$\frac{width-1.07}{b}$, b = lateral spacing required for a pedestrian to move (m), k and k_j = pedestrian density (peds/m^2) and maximum/jam density (peds/m^2), respectively, v_f = free flow speed of pedestrians (m/sec).

The model in equation (3.5) is empirically validated in Study [28] and we have selected this complete model to represent and update the speed of subjects in our simulator and consequently improve the credibility of our AVS solutions. In Chapter 5, we will illustrate the difference which was made through using this complete model in our AVS system.

To implement the following model, we have to carefully consider the required computational burden which will be caused by utilizing this inclusive speed model to update speed value of subjects, going through the surveillance site. As incorporating this speed model means a periodic call to the implemented model function, it is necessary to maximize the efficiency of the implemented model to have a minimum foot print in the total time requirements of our implemented simulator.

The included factorial function and sigma operation in the model makes it computationally challenging, especially for large amounts of s values. We have carefully examined the model in action and determined that large values of s, which is actually determined by the width of the surveillance area, have a small impact on the v_f speed. This is completely expected as having a large value for s means the surveillance site is more likely to be scarcely populated, meaning the subjects can move around the surveillance site, easily without causing a burden and hurdle for other subjects in the area.

We have determined that having a s value, greater than 21 have very small effect on the free-flow speed value of the present subjects in the surveillance site and this effect can be captured accurately enough by using a predetermined enormous number for factorial function and sigma operator. Considering this fact, we can immensely improve the efficiency of implementation by incorporating an optimized model, which for input s values greater than 21, considers this predetermined large number instead of mentioned challenging functions.

```
//Input: planSeedSet, a Set of Seed frames for each camera
//Output: ResultedPlan, Calculated plan, generated for all PTZ camera
BuildPlans(){
 for (each Camera cam){ // for every PTZ camera
  resultedPlan[cam] = {}; maximumRecognituinProbability = 0;
  // sort seed frames in every camera list based on the zoom value
  sort(planSeedSet[cam][ ], zoom);
  for (each seeds f){ //for every frame
   tempPlan = {}; planRecognitionProbability = 0; // initialize variables
   currentFrame = f; // set the starting frame node for current camera plan
   if (T_move,cam→currentFrame > T_pr) then
    continue; // The required time to move to frame currentFrame violates T_pr
   else {
    tempPlan.push(planSeedSet[cam][currentFrame]);
    planRecognitionProbability+ = ψ(currentFrame);
    forwardScanning = 1;
    T_rr = T_REC − T_e;
    while (T_rr > 0){
     if (currentFrame == firstSeedFrame) then forwardScanning = true;
     if (currentFrame == lastSeedFrame) then forwardScanning = false;
     if (forwardScanning == 1) then currentFrame + +;
     else currentFrame − −;
     if ((T_move,cam→currentFrame + T_e) < T_rr) then{
      tempPlan.push(planSeedSet[cam][currentFrame]);
      planRecognitionProbability+ = ψ(currentFrame);
      T_rr = T_rr − T_e;
     } End of if
    } End of while (T_rr > 0)
   } End of else T_move is valid
   if (planRecognitionProbability > maximumRecognitionProbability){
    maximumRecognitionProbability = planRecognitionProbability;
    resultedPlan[cam] = tempPlan;
    T_SLR = CurrentSystemLeadTimeForCamera(cam)
   } End of if
  } End of frame loop j
  if(T_MSLR < T_SLR)
   T_MSLR = T_SLR;
 } End of camera loop cam
 nextTrackTime = nextTrackTime − T_P + T_MSLR;
 AdjustT_P(T_eP, T_MSLR)
} // End of BuildPlans()
```

Figure 3.6: Enhanced EVB Simplified Procedure in Building the Scheduling Plan for Each Camera, [T_{eP}: previous spent pre-recording time, T_P: pre-recording phase time, T_{REC}: recording phase time, T_{pr}: unused available time in the pre-recording phase, T_e: time spent on recording one frame, T_{rr}: unscheduled available time in the recording phase, $T_{move,cam \rightarrow currentFrame}$: time to move camera cam to frame currentFrame. T_{SLR}: pre-recording time required for the best selected plan so far for one specific camera, T_{MSLR}: system pre-recording time required for the best selected plan for all the cameras]

Chapter 4: Performance Evaluation Methodology

We have developed in C++ a simulation application, called *AutoSurvSimEnhanced*, for an AVS system, which implements the proposed enhanced solution for scheduling the PTZ cameras in the surveillance site. We have assumed a surveillance site with a rectangular area, without losing the generality of the problem statement. Subjects enter the site from the four sides of the rectangular area, moving in a randomly selected direction of movement and a randomly selected initial speed. The speed later is adjusted continuously during the presence of the subjects in the area using the real and inclusive speed model, as it is necessary to consider the density of the subjects in the covered area to hold the integrity of the simulated solution being applied to the scheduling problem. Poisson, Gaussian, and uniform distributions are incorporated in order to model subject arrivals, initial speed, and the direction or angle values [2]. PTZ cameras are placed around the edges of the rectangular area of the surveillance site.

Each PTZ camera is also simulated by considering all the related specifications, including the PTZ movement speed, sensor dimension, and the focusing time. Furthermore, we have taken into account the impact of frame dependence as it was previously implemented.

Table 4.1: Subjects and Surveillance System Related Specifications

Parameter	**Value(s)**
Request Arrival Distribution	Poisson
Request Speed Distribution	Truncated Normal (mean=1.5 Request/second, std=0.5)
Site Area	Variable, Default = $80m \times 60m$
Mean Request Arrival Rate (μ)	2-38, Default = 30 Request/second
Entrance Angle	-40°- 40°with $\perp$ to entrance side
Number of Cameras	2-8, Default = 8
PTZ Tilt	0°-90°
PTZ Zoom	1-50 levels
PTZ Pan	0°-180°
Sensor Height Pixel Count	768 pixels
Sensor Width Pixel Count	1024 pixels
Focus Latency	0.5 sec
Pre-recording	2-15 sec, Default = 3
Recording Time	2-6 sec, Default = 3
Number of Selected Frames in the Ordered List	15
Independence Threshold	0.5 second
Cluster Width	Variable, Default=5.3m
Angle Threshold	25 degrees

Table 4.1 shows the main environment parameters and their values and summarizes the parameters used in the evaluation. The simulations were executed on a desktop PC with a 64-bit 3.6 GHz Quad-core CPU and 16 GB RAM. 6000 subjects were considered passing true the surveillance site throughout the time, before reporting and recording the average measured values.

We consider two performance metrics: the *subject recognition probability* and the *total algorithm time*.

More than 600 simulation scenarios were considered and more than 10000 simulation runs were executed to clearly depict the improvements, resulted by applying the proposed enhancements to the solutions. Each simulation case was executed multiple times to eradicates the effect of exceptional execution cases. Later, all the detailed statistics of the simulation results will be publicly accessible online.

4.1 Environment and System Parameters and Settings for Each Specific Set of Simulation Runs

Each specific group of simulation runs which were conducted to demonstrate different specifications and attributes of our enhanced solutions are described here in details to clearly specify the environment and system parameters, and subsequently make the reruns and repeatability of the experiments thoroughly achievable.

4.1.1 Number of PTZ Cameras Analysis

In this set of simulation runs, the number of cameras were changing from 2 to 8, incrementally and the portion of recognized subjects to total number of subjects, subject recognition probability and the average algorithm time were measured and reported to clearly differentiate between the performance of GBG and EBP scheduling methods. The numbers for BFG was discarded, as it was severely under-performing in comparison to the other two methods.

In Table 4.2 we can see all the important parameters and environment setting values, used in conducting the related simulation runs.

Table 4.2: Summary of Workload and System Characteristics in Examining the Effect of Number of PTZ Cameras

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	10 request/second
Clustering	Disabled
Dynamic Pre-Recording Time	Disabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Disabled
PTZ Cameras	Variable, 2-8
Pre-recording	2 seconds
Recording	2 seconds

4.1.2 Mean Subject Arrival Rate Analysis

In this group of simulation executions, the subject arrival rate was changing from 6 to 16, incrementally by adding 2 request/second in each step and the percent of covered subjects, subject recognition probability and the average algorithm time were examined to comprehensibly differentiate between the operational performance of GBG and EBP scheduling methods. The numbers for BFG, as it was the case with previous set of simulation runs, was discarded as it was very low in comparison to the other two methods.

In Table 4.3, the important parameters and environment setting values, used in conducting the related simulation runs are demonstrated.

Table 4.3: Summary of Workload and System Characteristics in Examining the Effect of Number of PTZ Cameras

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	Variable, 6-16 request/second (increased by 2 in each step)
Clustering	Disabled
Dynamic Pre-Recording Time	Disabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Disabled
PTZ Cameras	8
Pre-recording	2 seconds
Recording	2 seconds

4.1.3 Analysis of Clustering Method

In this sets of simulation runs, the number of PTZ cameras, the subject arrival rate and the site area dimensions were changing from 2 to 6, from 6 to 16, incrementally by adding 2 request/second in each step, and from 50x30 m^2 to 140x120 m^2, respectively. The subject recognition probability in addition to the average algorithm time were examined to completely demonstrate the benefits of applying a clustering approach and how it affects operational performance of GBG and EBP scheduling methods.

Tables 4.4, 4.5, and 4.6 demonstrate the important parameters and environment setting values, used in conducting the related simulation runs, in examining the clustering approach.

Table 4.4: Summary of Workload and System Characteristics in Examining the Effect of Clustering Approach by Utilizing Different Number of PTZ Cameras

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	14 request/second
Clustering	Enabled (under Examination)
Clustering Width	5.7 m
Dynamic Pre-Recording Time	Disabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Disabled
PTZ Cameras	Variable, 2-8
Pre-recording	2 seconds
Recording	2 seconds

Table 4.5: Subjects and Surveillance System Related Specifications in Examining the Effect of Clustering Approach by Different Subject Arrival Rates

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	Variable, 10-24 request/second (increased by 2 in each step)
Clustering	Enabled (under Examination)
Dynamic Pre-Recording Time	Disabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Disabled
PTZ Cameras	8
Pre-recording	2 seconds
Recording	2 seconds

Table 4.6: Summary of Workload and System Characteristics in Examining the Effect of Clustering Approach by Different Surveillance Site Areas

Parameter	**Value(s)**
Scene Area	50 × 30, 60 × 40, 70 × 50, 80 × 60, 90 × 70, 100 × 80, 110 × 90, 120 × 100, 120 × 100, 130 × 110, 140 × 120 m^2
Mean Request Arrival Rate	14 request/second
Clustering	Enabled (under Examination)
Dynamic Pre-Recording Time	Disabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Disabled
PTZ Cameras	8
Pre-recording	2 seconds
Recording	2 seconds

4.1.4 Cluster Size Analysis

This set of simulation runs, utilizes different maximum cluster Intra-Subject distances. This value changes from 0 (Disabled Clustering) to 4.8 meters, by 0.2 increments in each step and the subject recognition probability was examined to clearly demonstrate the effect of cluster size on the operational performance of GBG and EBP scheduling policies.

In Table 4.7, the important parameters and environment setting values, incorporated in executing the related simulation runs can be observed.

Table 4.7: Subjects and Surveillance System Related Specifications in Examining the Cluster Size Influence

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	14 request/second
Clustering	Enabled
Dynamic Pre-Recording Time	Disabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Disabled
PTZ Cameras	8
Pre-recording	2 seconds
Recording	2 seconds

4.1.5 Timing Analysis in Parallel Implementation of Scheduling Algorithms

In simulation runs to examine the implementation policies for the parallel scheduling algorithm, the mean subject arrival rate was changing from 2 to 38, incrementally and the computation times were measured for different components of the solution architecture to guide the decisions regarding an implementation plan for the parallel solution. The measured component units were clustering, frame generation and satisfaction calculation, frame filtration, and frame set selection and planning.

In Table 4.8, we can see all the important parameters and environment setting values, utilized to execute the related simulation runs.

Table 4.8: Subjects and Surveillance System Related Specifications in Examining the Calculation Time for Different Units of System

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	Variable, 2-38 request/second (increased by 2 in each step)
Clustering	Enabled
Dynamic Pre-Recording Time	Enabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Disabled
PTZ Cameras	8
Pre-recording	6 seconds
Recording	6 seconds

4.1.6 Analysis of Parallel — Dynamic Approach

These sets of simulation executions were created by changing the pre-recording time and subject arrival rate values, from 3 to 19 seconds and from 2 to 38 subjects per second, respectively and incrementally. The pre-recording value was incremented by 1, in each step, and the subject arrival rate was incremented by 2, in each step. Four different settings were considered, including, static pre-recording time plus sequential approach, static pre-recording time plus parallel approach, dynamic pre-recording time plus sequential approach, and dynamic pre-recording time plus parallel approach. The subject recognition probability and the average algorithm time were assessed to clearly emphasize the benefits, gained by utilizing our enhanced scheduling solutions.

In Tables 4.9 and 4.10, the important parameters and environment setting values, used in conducting the related simulation runs are presented.

Table 4.9: Summary of Workload and System Characteristics in Examining the Performance Benefits of Dynamic Parallel Approach under Different Pre-Recording Time Values

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	30 request/second
Clustering	Enabled
Dynamic Pre-Recording Time	Enabled (under Investigation)
Parallel Execution in Frame Generation and Satisfaction Calculation	Enabled (under Investigation)
PTZ Cameras	8
Pre-recording	Variable, 3-19 seconds
Recording	2 seconds

Table 4.10: Summary of Workload and System Characteristics in Examining the Performance Benefits of Dynamic Parallel Approach under Different Subject Arrival Rate Values

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	Variable, 2-38 request/second
Clustering	Enabled
Dynamic Pre-Recording Time	Enabled (under Investigation)
Parallel Execution in Frame Generation and Satisfaction Calculation	Enabled (under Investigation)
PTZ Cameras	8
Pre-recording	6 seconds
Recording	6 seconds

4.1.7 Analysis of New Speed Model

These groups of simulation runs were produced by changing the pre-recording time and subject arrival rate values, from 3 to 19 seconds and from 40 to 68 subjects per second, respectively and incrementally. The pre-recording value was incremented by 1, in each step, and the subject arrival rate was incremented by 2, in each step. Two different cases were considered, including, the constant speed model and the new inclusive speed model which considers many factors including the density of the surveillance site. The subject recognition probability and the average algorithm time were assessed to clearly emphasis the benefits gained by utilizing our enhanced scheduling solutions.

In Tables 4.11 and 4.12, the important parameters and environment setting values, used in conducting the related simulation runs are presented.

Table 4.11: Summary of Workload and System Characteristics in Examining the New Inclusive Speed Model under Different Pre-Recording Time Values

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	60 request/second
Clustering	Enabled
Dynamic Pre-Recording Time	Enabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Enabled
PTZ Cameras	8
Pre-recording	Variable, 3-19 seconds
Recording	2 seconds

Table 4.12: Summary of Workload and System Characteristics in Examining the New Inclusive Speed Model under Different Subject Arrival Rate Values

Parameter	**Value(s)**
Scene Area	$80m \times 60m$
Mean Request Arrival Rate	Variable, 40-68 request/second
Clustering	Enabled
Dynamic Pre-Recording Time	Enabled
Parallel Execution in Frame Generation and Satisfaction Calculation	Enabled
PTZ Cameras	8
Pre-recording	2 seconds
Recording	2 seconds

Chapter 5: Presentation and Analysis of Main Results

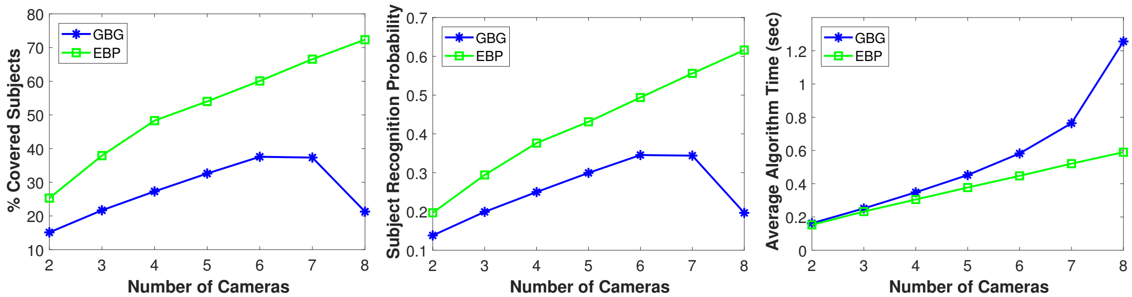
In this section, we compare the effectiveness of the three proposed schemes under different scenarios. Only the main results are shown.

5.1 Analysis of the Influence of Number of PTZ Cameras

Let us now compare the effectiveness of the two scheduling schemes as the number of PTZ cameras is varied. Figure 5.1 shows these results in terms of the percentage of covered subjects, average subject recognition probability, and average algorithm time. EBP performs significantly better than GBG in each one of the three metrics. This performance enhancement in EBP is attributed to allowing each camera to view a different set of subjects in a recording phase as oppose to GBG which constantly covers the same frame during the recording phase. The enhanced performance in time complexity, however, is due to reducing the search space significantly by selecting only the best combination of seed frames for each camera. The BFG method is not demonstrated in these figures as it was significantly lower than the other two algorithms and it was not able to complete the simulation tasks in a conceivable time. The ratio of covered subjects, as well as the average subject recognition probability improves with the incremented number of PTZ cameras. At the same time, it increases the time complexity. These performance metrics in GBG decreases after the number of employed cameras reaches 6 as the time complexity increases significantly, enforcing the system to prematurely abort the pre-recording period, resulting in poor performance in recording phase. The higher time complexity is due to the increase in the search domain, as additional FoRs have to be searched to get the best frame sets. Quantity of deployed PTZ cameras should be selected based on the performance-time tradeoff. Highest performance belongs to the EBP scheduling scheme while it induces the least time complexity.

5.2 Analysis of Influence of Mean Subject Arrival Rate

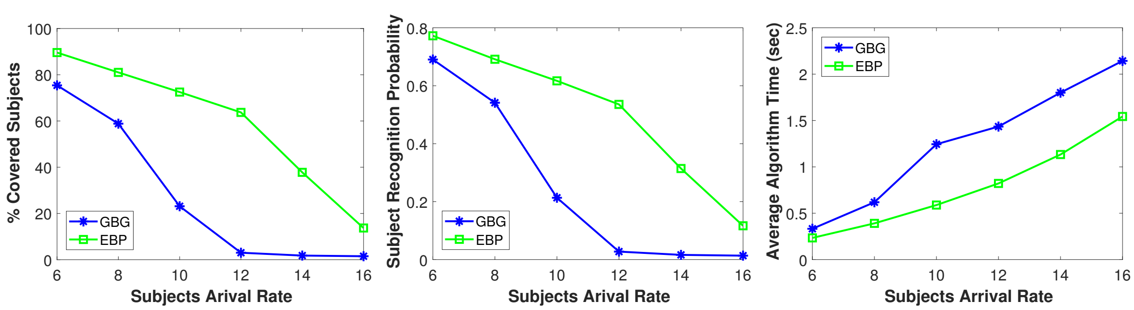
Different scheduling schemes performs differently by increasing the mean subject arrival rate. This fact is shown in Figure 5.2. The figure demonstrates the performance differences of the EBP and GBG



(a) Percentage of Covered Subjects (b) Average Subject Recognition Prob. (c) Time Complexity

Figure 5.1: Comparing Different Schemes for Different Number of PTZ Cameras

scheduling schemes by increasing the subject arrival rates. As expected, the three metrics become worse because when the arrival rate is increasing, there is a higher chance a subject departs the surveillance area without being captured by a camera and subsequently, the spent time on the captured subjects will decrease, resulting in a decreased performance. As we can see in this figure, performance of EBP method is much greater than GBG when high arrival rate values should be handled, and that the difference in performance and execution time between EBP and GBG schemes becomes more pronounced as the arrival rate increases. Because of the sweeping movements of the EBP in recording periods, a higher number of subjects will be captured as oppose to GBG method which needs to finish the current recording period by constantly covering the same frame and then regenerates a new plan which would be covered in the next recording period. As previously mentioned in 5.1, the related results for BFG is not demonstrated here for the same mentioned reason.



(a) Percentage of Covered Subjects (b) Average Subject Recognition Prob. (c) Time Complexity

Figure 5.2: Comparing Different Schemes for Different Mean Subject Arrival Rates

5.3 Effect of Applying Clustering Approach

5.3.1 Clustering Influence under Different Number of Cameras

In Figure 5.4(a) is demonstrated that by increasing the number of PTZ cameras, average subject recognition probability will increase. EBP-C produces the best overall performance. The other schemes (GBG, GBG-C, and EBP) have higher computational complexity, and thus after increasing the number of cameras beyond a certain threshold value, they fail to improve the recognition probability as they are unable to finish the required algorithmic computations before the beginning of the recording period. The time complexities of the algorithms are analyzed in Figures 5.4(b) and 5.4(c). By increasing the number of cameras, the computational complexities and thus the algorithm times increase. As it can be seen from these figures, EBP-C and GBG-C have the lowest computational complexity.

5.3.2 Clustering Influence under Different Subject Arrival Rate

Figure 5.5(a) illustrates the effect of subject arrival rate on the average subject recognition probability. GBG-C and EBP-C perform much better than GBG and EBP, respectively, when the arrival rate is increasing.

When the clustering method is incorporated in the surveillance system, the element of scalability increases as it would decrease the computational complexity by only concentrating on the dense areas of the surveillance site. EBP-C behaves better when the arrival rate is high. The average number of subjects that are simultaneously present in the surveillance area increases with the subject arrival rate, assuming that the area is fixed. Consequently, the probability of a subject being occluded by other subjects increases, thereby worsening the recognition probability. This trend persists until the arrival rate is high enough to the extent that most of the subjects lose any chance of being covered by any camera, and thus the recognition probability becomes practically zero. Figures 5.5(b) and 5.5(c) show how the time complexities increase with the arrival rate. This behavior is expected as higher subject arrival rates lead to a more crowded surveillance site, thereby complicating the pre-recording calculations of the algorithm. EBP-C and GBG-C have much lower computational times than EBP and GBG, respectively.

5.3.3 Clustering Influence under Different Surveillance Site Areas

Figure 5.6(a) shows how increasing the area of the surveillance site affects the subject recognition probability. As the surveillance site area increases, PTZ cameras cannot cover the entire area as effective as they were able to do so in a small surveillance site. The initial small increase in the EBP-C is due to subject concentration in the site area. In EBP-C, the first three site dimensions are not spacious enough and the subjects are too close to each other. The clustering process in this condition will not be effective as the objects inside a cluster will probably cover each other completely and degrade any chance of subject recognition by the PTZ cameras. Therefore, by increasing the site area, there is an initial improvement in the results gained by clustering. As shown in Figures 5.6(b) and 5.6(c), By increasing the area of the surveillance site, the average algorithm time increases as it takes a longer time for a subject to departure the site and subsequently it will contribute more to the computational complexity of the algorithm.

Finally, Figure 5.3 illustrates the impact of the cluster size. By increasing the cluster size, more subjects can be covered, thereby achieving better performance. This performance enhancement after a certain value will stop as the resolution of the captured subjects in the cluster will decrease and subsequently the benefits of having a larger cluster size are canceled out by the lower resolutions.

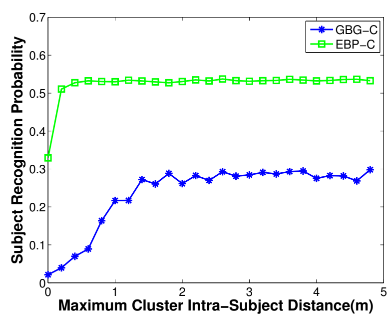
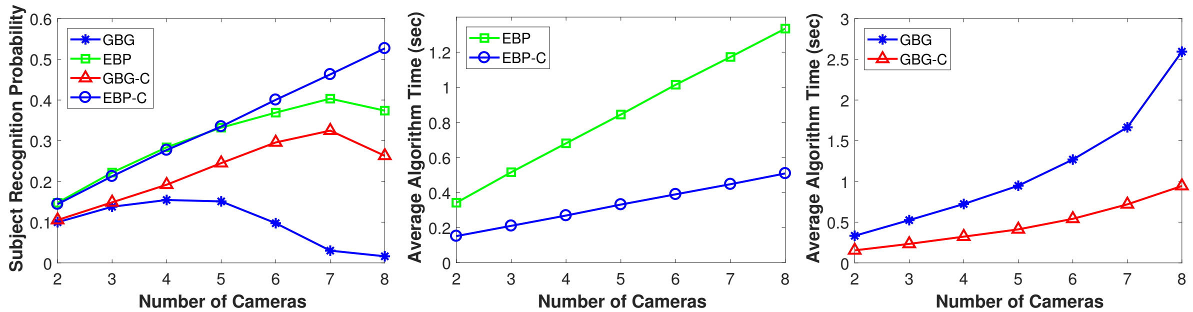


Figure 5.3: Effectiveness of the Clustering Approach by Varying the Cluster Size

5.4 Time Analysis

In order to implement a parallel algorithm for the PTZ camera control problem, the AVS system was divided to different operational units, including clustering, frame generation and satisfaction calculation,

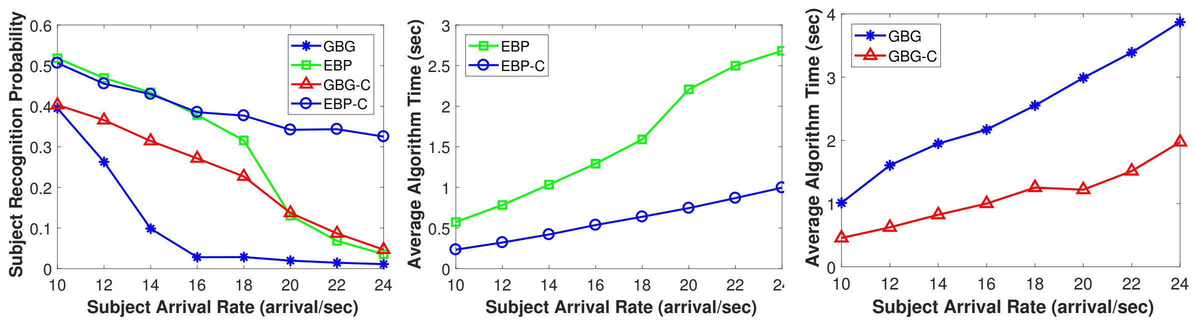


(a) Average Subject Recognition Probability

(b) Average Time

(c) Average Time

Figure 5.4: Comparing Impact of Clustering by Varying the Quantity of PTZ Cameras



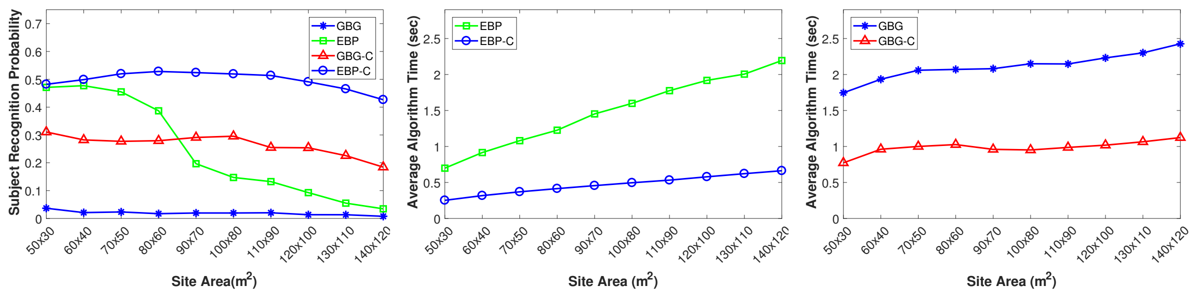
(a) Average Subject Recognition Probability

(b) Average Time

(c) Average Time

Figure 5.5: Comparing Impact of Clustering by Varying the Mean Arrival Rate

frame filtration, and frame set selection and planning. All the units were timed multiple times with different system characteristics and all the units were initially implemented as parallel units in order to determine if the synchronization and communication costs of the operational units will be compensated by the gain in performance advantages of simultaneous threads of execution. The granularity of parallel implementation was selected as the number of cameras in the AVS system which results in a perfect scalability of the system with the increment of number of cameras. It was determined that by implementing only the Frame satisfaction calculation unit as a parallel algorithm, the performance advantage will be gained and the synchronization and communication costs will be smaller than the performance gain in simultaneous execution. Figure 5.7 illustrates the computation time for different units of the AVS system. The frame satisfaction unit has the highest time requirement and complexity and the difference will increase by increasing the subject



(a) Average Subject Recognition Probability

(b) Average Time

(c) Average Time

Figure 5.6: Comparing the Impact of Clustering by Varying the Area of The Surveillance Site

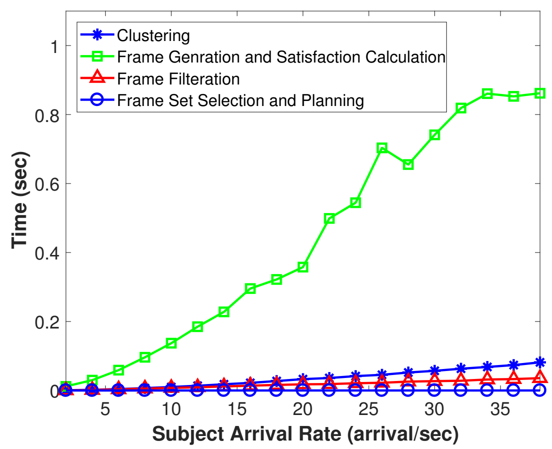
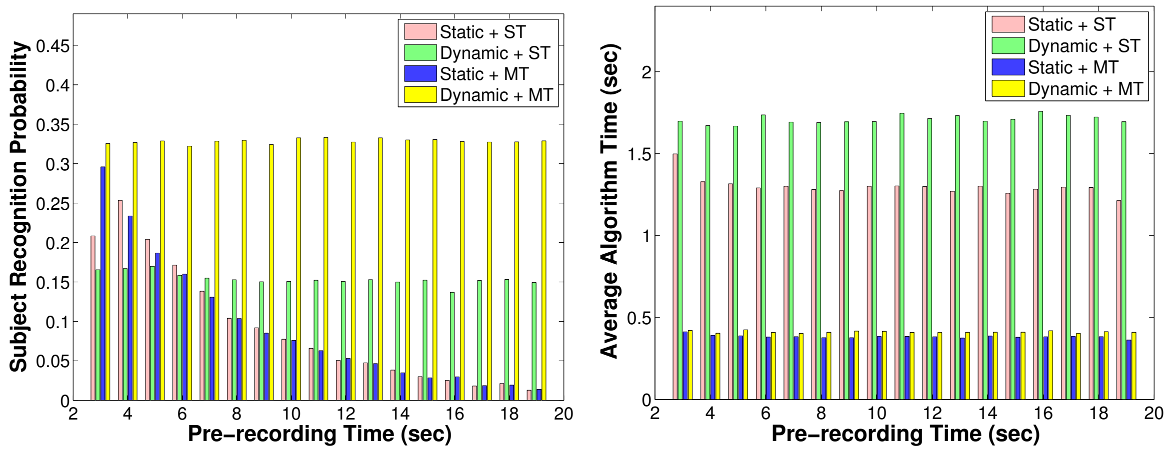


Figure 5.7: Computation Times for Different Modules of the Control Solution

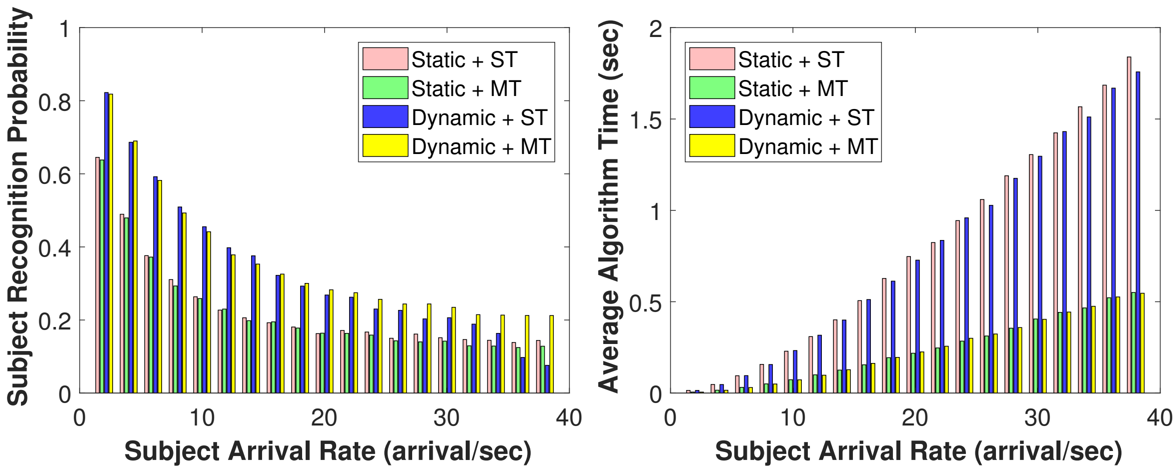
arrival rate. Therefore, this unit is the perfect candidate for a parallel implementation.

Figure 5.8 demonstrates the effect of pre-recording time on subject recognition probability and average algorithm time in four different cases. A sequential static solution versus a parallel one and similarly a dynamic sequential solution against a parallel implementation. The dynamic parallel solution in Figure 5.8(a) consistently outperforms other methods in subject recognition probability. The performance in dynamic sequential setting is steady but in comparison to the parallel case falls short as the algorithm is more time consuming in the sequential case which in turn decreases the chance of an early complete calculation in the pre-recording period. The average algorithm time for parallel pair solutions are significantly lower than the sequential pair as Figure 5.8(b) indicates. Although the static solutions have a lower average algorithm time, it should be considered that in static cases, the number of subjects who passes through the surveillance area without getting sufficiently recognized and getting involved in different processes and calculations, are significantly lower than that of the dynamic cases and as a result the average calculation complexity in the dynamic cases are higher in comparison to static algorithms. This fact can clearly be recognized, as the difference between the subject recognition probability of dynamic and static cases are immense in large pre-recording time values.



(a) Subject Recognition Probability (b) Total Calculation Time

Figure 5.8: Comparing Effectiveness of Dynamic Parallel Algorithm in Subject Recognition Probability and Average Algorithm time with Pre-recording Time [Subject Arrival Rate = 30 request/sec]



(a) Subject Recognition Probability (b) Total Calculation Time

Figure 5.9: Comparing Effectiveness of Dynamic Parallel Algorithm in Average Subject Recognition Probability and Average Algorithm Time with Subject Arrival Rate [Clustering is enabled, Pre-recording: 2s, Recording: 2s]

Figure 5.9 demonstrates the effect of subject arrival rate on subject recognition probability and average algorithm time in four different scenarios. The dynamic solution pair generally outperform the static pair as it was the case with the Figure 5.8(a). Here in Figure 5.9(a), the dynamic parallel and sequential algorithms perform almost similarly until the arrival rate reaches a threshold value of 16 arrival/sec. After this point, the dynamic parallel solution significantly outperforms the sequential one. This is because before this threshold value, the computational complexity of operations, performed during the pre-recording time was not challenging enough to favor the parallel implementation over the sequential one. There is almost no difference in performance between the static pair solutions. This is completely expected as both options will not use any available spare time in the pre-recording time, even if one is finished sooner than the allowed time.

5.5 Effect of Incorporating the New Speed Model

Let us now demonstrate the effect of using the new realistic speed model for subjects in the surveillance site. Figure 5.10 illustrates the effect of using the new model over the constant model in subject recognition probability and total algorithm time, against different subject arrival rates. Using the new realistic speed model yields a more accurate and smaller recognition probability, as the subjects in the constant speed model are passing through each other continuously, causing many reappearing of the face surfaces with respect to a specific camera and contributing to a higher recognition probability for the subject. On the other hand, the realistic new speed model causes the subjects to slow down in the dense surveillance site and having a lower chance to have their faces to be covered by a camera's field of view as there is a higher chance the subjects would stock behind other subjects because of this hindered movement.

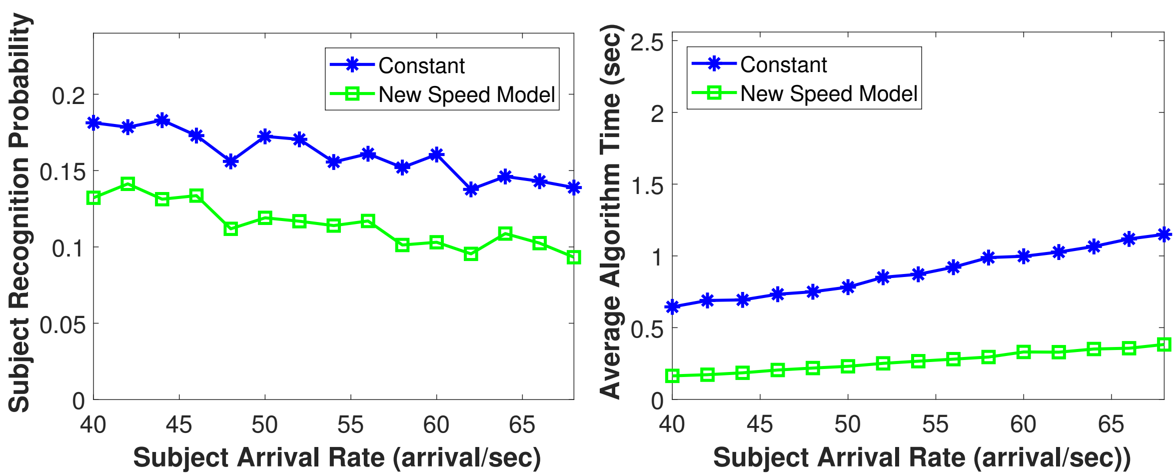
By increasing the subject arrival rate, there is smaller probability for each subject to be covered by a PTZ camera, resulting in a smaller recognition probability as Figure 5.10(a) demonstrates.

The algorithm time for the new speed model is significantly lower in comparison to the constant model as it is demonstrated in Figure 5.10(b). The reason is the hindered movement, realized by the realistic speed model, resulting in a lower chance for a subject to be uncovered during the movements in the surveillance site and subsequently contributing to a less complex computation in determining the plan for each camera.

Figure 5.11 shows the effect of using the new model over the constant model, in subject recognition probability and total algorithm time against different pre-recording values. As the dynamic pre-recording method was used in the setup, the subject recognition probability and average algorithm time is almost constant in both the constant and new speed models across the chart, which is completely expected as the unused time in a pre-recording period would be used to schedule an earlier recording and tracking period.

Using the new realistic speed model yields a more accurate and smaller recognition probability as it appears in Figure 5.11(a). The reason is similar to the explanation in Figure 5.10(a).

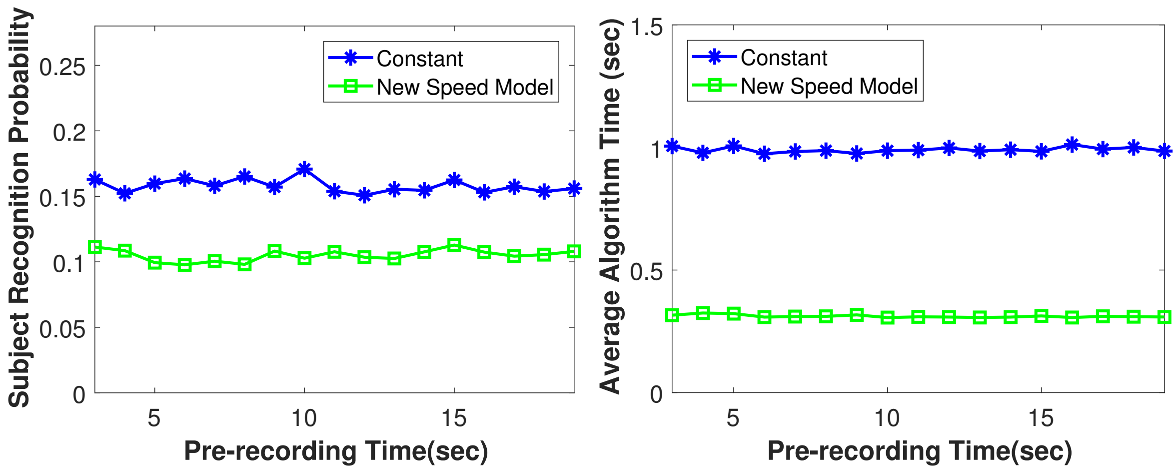
Similarly, Figure 5.11(b) shows the algorithm time for the new speed model is significantly lower in comparison to the constant model for the same reason, previously mentioned in explanation for Figure 5.10(b).



(a) Subject Recognition Probability

(b) Total Calculation Time

Figure 5.10: Comparing the New Realistic Speed Model and Constant Model in Subject Recognition Probability and Average Algorithm Time with Varying Subject Arrival Rates [Pre-Recording Time = 2 sec, Recording Time = 2 sec]



(a) Subject Recognition Probability (b) Total Calculation Time

Figure 5.11: Comparing the New Realistic Speed Model and Constant Model in Subject Recognition Probability and Average Algorithm Time with Varying Pre-Recording Times [Subject Arrival Rate = 60 subjects/sec]

Chapter 6: Conclusions and Future Work

We have designed and developed an enhanced scheduling solution for the PTZ camera control problem. The solution optimizes the average subject recognition probability. The solution includes a parallel implementation of the frame satisfaction calculation which scales perfectly by the number of utilized PTZ cameras. This parallel approach, which can be used in conjunction with various camera scheduling schemes, achieves significant improvements in the subject recognition probability and the algorithm computation time.

Additionally, in order to take advantage of this speed enhancement, we have proposed and implemented a dynamic pre-recording method which tries to salvage the unused time portion of the pre-recording phase to schedule the recording stage sooner, increasing the efficiency and in turn increasing the recognition probability of the surveillance system. Also, we have incorporated a realistic, inclusive speed model which uses many factors, including social elements, to realistically update the subjects speed value in the surveillance site, increasing the reliability of the proposed enhanced solution.

A clustering method has been incorporated in our system as a pre-step to frame generation and satisfaction calculation, resulting in populating frames more effectively. Subsequently, it enables us to focus on the areas that are populated with subjects. We have also analyzed the effectiveness of clustering on the performance of camera scheduling in terms of recognition probability and the average algorithm time, considering the impacts of several major parameters.

Our parallel implementation of the PTZ camera scheduling solution empowers the system to significantly take advantage of the multi-core processing units and as a result finishes the pre-recording required tasks much faster than the sequential implementations. By enabling the system to finish the computations sooner, taking advantage of the remaining time in the pre-recording phase becomes a necessity and this is where our dynamic pre-recording approach appears and by scheduling the recording phase sooner in comparison to the static method, increases the efficiency and in turn increases the subject recognition probability.

Our utilized new inclusive speed model considers the density of the subjects, present in the surveillance site which is a factor influenced by many elements, including social behavior of the subjects. By using this comprehensive speed model, we are increasing the credibility of our proposed enhanced solution as it updates the speed of subjects in the surveillance site, based on a realistic speed model.

Our proposed enhanced solution has been assessed by conducting extensive simulation (more than 10000 simulation runs). The main results can be itemized as follows.

- The dynamic pre-recording method enables high performance by efficiently using the required portion of the pre-recording period, enabling the system to schedule the recording period sooner, which makes the surveillance system more responsive and efficient in recognition of the subjects, passing through the surveillance site.

- A parallel implementation on the order of number of PTZ cameras significantly improves the calculation speed and in turn the scalability factor, as increasing the size of the surveillance system dictates an increase in the quantity of employed PTZ cameras.

- By applying a realistic speed model which is aware of population density impact on the speed of subject movements in the area, generating more realistic and credible results, especially in dense surveillance site settings is achieved as it is necessary to consider the speed of subjects in the surveillance site as they pass through the system.

- Our enhanced EBP PTZ camera scheduling scheme produces the highest recognition probability as it is the enhanced version of the best already available scheduling method.

- The increment in the quantity of PTZ cameras utilized in the surveillance system, has a sizable influence on the scheduling approaches. Different approaches can take advantage of the increased number of cameras at different rates.

- By increasing the number of deployed PTZ cameras, the subject recognition probability increases. At the same time, the plan building complexity becomes larger. Consequently, a tradeoff decision must be made if deploying more processing units, which can scale well by number of cameras, is not a tangible option.

- When the subject arrival rate is incremented or the area of the site being monitored is widen, EBP can produces the best performance and subsequently, the highest scalability as it requires the minimum computation time in comparison to other methods.

- The recognition probability and the required computation time are significantly improved by applying the clustering approach as a pre-step in PTZ camera scheduling schemes since it makes it possible for each PTZ camera to cover a larger number of subjects in the surveillance site, increasing the chance of a subject to be captured more in the site as it is crossing thorough the system.

- The recognition probability increases with the cluster size. The increase in performance with the cluster size saturates after a certain value because in larger clusters, subjects are seized at a lower resolution, and eventually the advantages of having a larger cluster size are neutralized by the lower resolutions.

- Increasing the subject arrival rate reduces the subject recognition probability. The reason, is by increasing the arrival rate, there is a higher chance to have a more populated and dense surveillance site, decreasing the probability of having a specific subject to be captured properly in a PTZ camera field of view. At the same time by having more subjects in the surveillance site, the computational complexity of the pre-recording phase will be higher, forcing the system to use more available time in this period, decreasing the probability of an early schedule to begin recording. This means the system will not be able to take advantage of the dynamic pre-recording method, as much as it is possible to do so by having a lower subject arrival rate.

- The chance of covering a subject with a high-resolution frame decreases as the site area is widened, causing a decrease in the recognition probability.

In the future, developing a new scheduling model by using a genetic algorithm in determining the order of traversed seed frames in plan building phase is of interest. Also, applying the enhanced solutions to a real surveillance system including physical PTZ cameras are highly desired (the implementation of the actual surveillance architecture). This system will include a network of high-end PTZ cameras, a network of wide-angle cameras, a multi-core processing architecture, high-speed network attached storage, and a highly configurable wireless router.

REFERENCES

[1] MarketsandMarkets, "Video surveillance market by system (analog, ip, biometrics), hardware (camera, monitors, servers, storage devices), software (video analytics, vms), and service (vsaas, installation and maintenance), vertical, and region - global forecast to 2022," 2017.

[2] Yiliang Xu and Dezhen Song, "Systems and algorithms for autonomous and scalable crowd surveillance using robotic PTZ cameras assisted by a wide-angle camera," *AUTONOMOUS ROBOTS*, vol. 29, no. 1, pp. 53–66, 2010.

[3] Hazem El-Alfy, David Jacobs, Larry Davis, and Larry Davis, "Assigning cameras to subjects in video surveillance systems.," in *proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 837–843.

[4] S. Lim N. Krahnstoever, T. Yu and K. Patwardhan, "Collaborative control of active cameras in large-scale surveillance," in *Networks Principles and Applications in Multi-Camera*, H. Aghajan and A. Cavallaro, Eds., pp. 165–188. Elsevier, 2009.

[5] M. S. Al-Hadrusi and N. J. Sarhan, "Efficient control of PTZ cameras in automated video surveillance systems," in *Multimedia (ISM), 2012 in Proceeding of IEEE International Symposium on*, Dec 2012, pp. 356–359.

[6] M. S. Al-Hadrusi, N. J. Sarhan, and S. G. Davani, "A clustering approach for controlling ptz cameras in automated video surveillance," in *2016 proceedings of the IEEE International Symposium on Multimedia (ISM)*, Dec 2016, pp. 333–336.

[7] http://www.bayometric.com/products/Face Recognition-SDK.htm, "Faceit SDK," .

[8] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Surveys*, vol. 35, pp. 399–458, December 2003.

[9] http://www.neurotechnology.com/face biometrics.html, “Verilook sdk,” .

[10] R.-L.V. Hsu, J. Shah, and B. Martin, “Quality assessment of facial images,” in *proceedings of Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, 19 2006-aug. 21 2006, pp. 1 –6.

[11] Ralph Gross, Simon Baker, Iain Matthews, and Takeo Kanade, “Face recognition across pose and illumination,” *Handbook of Face Recognition*, 2004.

[12] Yi Yao, Besma R. Abidi, Nathan D. Kalka, Natalia A. Schmid, and Mongi A. Abidi, “Improving long range and high magnification face recognition: Database acquisition, evaluation, and enhancement,” *Computer Vision and Image Understanding*, vol. 111, pp. 111–125, August 2008.

[13] Zihan Zhou, A. Wagner, H. Mobahi, J. Wright, and Yi Ma, “Face recognition with contiguous occlusion using markov random fields,” in *proceedings of IEEE International Conference on Computer Vision*, 29 2009-oct. 2 2009, pp. 1050 –1057.

[14] E. Harwood, *DIGITAL CCTV: A Security Professionals Guide*, Elsevier, 2008.

[15] H. Kruegle, *CCTV Surveillance: Analog and Digital Video Practices and Technology*, Elsevier, second edition, 2007.

[16] F. Nilsson, *Intelligent Network Video: Understanding Modern Video Surveillance Systems*, CRC Press, 2009.

[17] Xuhui Zhou, Robert T. Collins, Takeo Kanade, and Peter Metes, “A master-slave system to acquire biometric imagery of humans at distance,” in *proceedings of ACM SIGMM International Workshop on Video surveillance (IWVS)*, New York, NY, USA, 2003, pp. 113–120, ACM.

[18] Arun Hampapur, Sharath Pankanti, Andrew W. Senior, Ying li Tian, Lisa M. G. Brown, Ruud M. Bolle, and Ruud M. Bolle, “Face cataloger: Multi-scale imaging for relating identity to location.,”

in *proceedings of IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, 2003, pp. 13–20.

[19] Cash J. Costello, Christopher P. Diehl, Amit Banerjee, and Hesky Fisher, "Scheduling an active camera to observe people," in *proceedings of ACM International Workshop on Video Surveillance and Sensor Networks (VSSN)*, New York, NY, USA, 2004, pp. 39–45, ACM.

[20] Faisal Qureshi and Demetri Terzopoulos, "Surveillance camera scheduling: a virtual vision approach," *Multimedia Systems*, vol. 12, no. 3, pp. 269–283, 2006.

[21] Joe H. Ward, "Hierarchical grouping to optimize an objective function," *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 236–244, 1963.

[22] B. Schiele P. Dollar, C. Wojek and P. Perona, "Pedestrian detection: A benchmark," in *proceedings of Computer Vision and Pattern Recognition conference (CVPR)*, June 2009.

[23] J. Ponce D. A. Forsyth, *Computer Vision: A Modern Approach*, Prentice Hall, first edition, 2002.

[24] Mubarak Shah, "Fundamentals of computer vision," *University of Central Florida*, 1997.

[25] B.J. Boom, G.M. Beumer, L.J. Spreeuwers, and R.N.J. Veldhuis, "The effect of image resolution on the performance of a face recognition system," in *proceedings of International Conference on Control, Automation, Robotics and Vision,*, dec. 2006, pp. 1 –6.

[26] Jingdong Wang, Changshui Zhang, and Heung yeung Shum, "Face image resolution versus face recognition performance based on two global methods," in *proceedings of Asia Conference on Computer Vision (ACCV)*, 2004.

[27] Washington D.C. Bureau of Public Roads, U.S. Department of Commerce, "Bpr traffic assignment manual," 1964.

[28] Khalidur Rahman, Noraida Abdul Ghani, Anton Abdulbasah Kamil, Adli Mustafa, and Md Ahmed Kabir Chowdhury, "Modelling pedestrian travel time and the design of facilities: A queuing approach," *PloS one*, vol. 8, no. 5, pp. e63503, 2013.

ABSTRACT

TOWARDS OPTIMAL PTZ CAMERA SCHEDULING

by

SINA G. DAVANI

December 2017

Adviser: Dr. Nabil Sarhan

Major: Computer Engineering

Degree: Master of Science

The automatic control of Pan/Tilt/Zoom (PTZ) cameras has been a major research problem. We consider the control of PTZ cameras in a manner that optimizes the overall recognition accuracy. The camera control solution operates into two alternating phases: pre-recording and recording. In the first phase, the processing architecture performs the necessary algorithmic calculations to determine the optimal PTZ camera setting. However, in the second phase, the PTZ cameras apply the desired settings, capture the videos, and stream these videos to the proxy station for analysis.

We enhance the overall recognition accuracy by developing a parallel PTZ control algorithm, which reduces the time spent on pre-recording and thus increases the fraction of time dedicated to capturing the actual videos of the surveillance site. Additionally, we propose a dynamic approach for determining the pre-recording time and thus allowing the system to extract the best benefits of the parallel algorithm. As the parallel algorithm leads to early completion of the pre-recording tasks, the dynamic approach empowers the system to benefit from the unused remaining time in the pre-recording phase and subsequently to place more dedication to the actual recording.

We analyze the effectiveness of the proposed solutions through extensive simulation, considering the impacts of major parameters, including the subject arrival rate, surveillance area, and the number of cameras. To make the simulations as realistic as possible, we incorporate an inclusive speed model to constantly

update and maintain the speed values for related subjects while they are crossing throughout the surveillance site. This speed model considers many factors, including the social tendencies and density of the people present in the surveillance site. Our overall solution assumes realistic 3D environments and not just 2D scenes.

We demonstrate that the proposed parallel algorithm substantially reduces the pre-recording time. We also show that the combination of the proposed parallel algorithm and dynamic approach greatly enhances the overall face recognition accuracy.

AUTOBIOGRAPHICAL STATEMENT

Sina G. Davani is an ADAS Associate Software Engineer at CONTINENTAL AG. While working in the automotive industry he has been tasked with developing software solutions for advanced driver assistance systems. He received his B.S. degree in Computer Engineering from Isfahan University of Technology in 2012. His main research interests are automated video surveillance, systems simulation, bandwidth adaptations for video streaming systems, machine learning, computer vision, and parallel and distributed system design and implementation. He is a Ph.D. candidate in the ECE Department at Wayne State University.