A Clustering Approach for Controlling PTZ Cameras in Automated Video Surveillance

Musab S. Al-Hadrusi Nabil J. Sarhan Sina G. Davani

Media Research Lab, Dept. of Electrical and Computer Engineering
Wayne State University Detroit, Michigan 48202
Email: {hadrusi, nabil, sina}@wayne.edu

***Abstract*—The efficient control of Pan/Tilt/Zoom (PTZ) cameras has been a major research problem. This paper presents a solution that seeks to optimize the overall subject recognition probability by controlling various deployed cameras, based on the characteristics of the subjects in the surveillance area. In particular, we propose and analyze a clustering-based approach, which can be used in conjunction with recently proposed camera scheduling schemes, to achieve significant improvements in both the subject recognition probability and the algorithm computation time. We extensively analyze the effectiveness of the clustering approach, considering the impacts of subject arrival rate.**

***Index Terms*—Automated Video Surveillance, Face Recognition, PTZ Camera Control, PTZ Camera Scheduling.**

I. Introduction

High-end video surveillance systems employ Pan/Tilt/Zoom (PTZ) cameras, which offer a high degree of flexibility. Since the number of subjects is usually much larger than the number of video cameras, the problem to be addressed is how to assign subjects to these cameras. Numerous studies have dealt with this control problem [1], [2], [3], [4] (and references within). Unfortunately, these studies had many limiting assumptions and considered a small subsets of the factors. The overwhelming majority of these studies analyzed metrics, such as the captured resolution, the percentage of subjects covered, and the number of times a subject is viewed, but the main objective in AVS systems should be maximizing the overall threat or subject detection/recognition probability.

This paper develops an overall solution for controlling PTZ cameras in such a way that optimizes the overall subject/object recognition probability. The control of cameras is based on many factors, such as the direction of the subject's movement and its location, distance from the cameras, occlusion, overall recognition probability so far, and the expected time to leave the site, as well as the movements of cameras and their capabilities and limitations. This paper builds on our preliminary work in [5] by proposing and analyzing a clustering-based approach, which can be used in conjunction with the camera scheduling schemes in [5], to achieve significant improvements in the subject recognition probability and the algorithm computation time. The main contributions of the overall solution can be summarized as follows: (i) optimizing the overall subject recognition probability by employing efficient clustering of subjects and considering the impacts of a large number of influences on this probability, (ii) addressing the camera scheduling problem in realistic 3D environments, and (iii) incorporating the practical limits and capabilities of the PTZ cameras.

This work was supported in part by U.S. NSF grant CNS-0834537.

We focus primarily on the face recognition algorithm and use its accuracy as the metric guiding the camera controlling process. We analyze the effectiveness of clustering on camera scheduling schemes in terms of the recognition probability and average algorithm time, considering the impact of subject arrival rate.

The rest of this paper is organized as follows. Section II discusses the background information and related work. Section III presents the proposed solution. Section IV discusses the performance evaluation methodology. Finally, Section V presents the main results.

II. Background and Related Work

Recent research has attempted to utilize the potential power and flexibility of PTZ cameras. Distant Human Identification (DHID) system [6] illustrated a typical master-slave system configuration. A fixed wide angle camera is used to observe a scene and send information to a server, which in turn analyzes the scene and sends commands back to PTZ cameras. The PTZ cameras capture finer frames for targeted subjects. Each camera is assigned to exactly one subject at a time.

Choosing which subject to capture next is determined by the scheduling policy. Different scheduling policies adopt different objectives. Paper [7] scheduled subjects by using precalculated plans based on a utility function, with each plan having a list of subjects to record at different times. Subject grouping was not considered in this work, and the results considered only people tracking objective.

Subject grouping is another technique used to tackle the scalability problem by grouping more than one subject in the same frame and assigning the frame to only one PTZ camera. Paper [2] modeled the surveillance system as a bipartite graph. The subject-to-camera assignment was obtained by applying the maximum matching algorithm. Grouping is based on the spatial scene and is controlled by modifying the radius of a disc centered around each subject. Subsequently, the sets are mapped to cameras. The work did not provide any specific function to arbitrate between frames or groups in the scheduling process.

Paper [1] used a master-slave system with a wide angle camera employed to collect scene information and pass it to the processing system. The processing system scans the data and runs a *Lattice-Based Grouping* (LBG) algorithm to generate a set of frames that represent the maximum "Objective Satisfaction" based on weighted subject resolution and coverage. The paper assumed that all PTZ cameras have

exactly the same field-of-regard (FoR) and can cover the scene equally likely. This assumption oversimplifies the problem and greatly reduces the candidate frames search domain. In addition, the frames in the search domain are generated from a 2D environment. Moreover, the quality of the captured frames did not include information about subject's pose or occlusion, thereby complicating the face recognition task. The work addressed a generic objective and showed the results for grouping in a tracking application. Further, the "subject satisfaction" was calculated in a linearly-additive fashion, without considering the probabilistic nature of the recognition process.

In the preliminary work in [5], we developed a solution for optimizing the overall subject recognition probability by controlling the pan, tilt, and zoom of various deployed PTZ cameras. In contrast with prior solutions, the developed solution considers the recognition probability, deals with cameras having different FoRs, generates frames according to cameras capabilities and not site area dimensions, and assesses the overlap among frames in 3D scenes. We developed two schemes for scheduling the PTZ cameras: *Grid-Based Grouping* (GBG) and *Elevator-Based Planning* (EBP). GBG is an adaptation of the LBG scheme in [1], incorporating the aforementioned enhancements. In contrast, EBP refines the filtered frames and uses them to build detailed plans for each camera. The scheme captures the current settings of each camera, and its moving, zooming, and focusing speeds. Thus, EBP allows each camera to view a different set of subjects at different times during one recording period rather than repeatedly tracking the same subjects. The main contribution of this new paper is proposing and analyzing 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.

III. Proposed Solution

A. Overview of the Overall Solution

The employed AVS system employs wide-angle cameras, PTZ cameras, and a processing architecture. The system operates into two alternating periods: a *pre-recording period* of T_p seconds and a *recording period* of T_R seconds, as in [1]. In the first period, the processing system reads the state and performs predictions and algorithmic calculations to assign frames to PTZ cameras. In the second period, the processing system starts receiving high quality frames from the PTZ cameras and considers them for further processing. During that recording period, the PTZ cameras track the subjects.

The solution targets the camera scheduling and assignment problem in 3D environments. Its main objective is to optimize the overall subject recognition probability by controlling the pan, tilt, and zoom of various deployed PTZ cameras, based on the characteristics of the subjects in the surveillance area. This control of cameras is based on the previously mentioned characteristics of the system.

The solution consists of four phases.

Frame Generation– The processing system utilizes the frames captured by the fixed wide-angle cameras to detect subjects and determine their attributes, including location, speed, and direction of movement, using one of the widely researched pedestrian detection algorithms [8]. Subsequently, the processing system predicts the future locations of the subjects when the PTZ cameras enter the recording and tracking phase. It then uses the prediction results to generate the set of all possible frames that can be captured by each PTZ camera through the examination of the different combinations of all camera settings. Clustering is used to group subjects, as will be shown later. The frame here is basically the *projection* of the view as seen by the camera at a specific field-of-view (FoV). By examining different settings, a set of frames encompassing the entire fields-of-regard (FoR) will be generated. Combining all possible frames from all cameras produces the entire frame domain that will need to be analyzed in later phases. A projection triangulation is used to estimate the 2D frame characteristics from the 3D site, and rotation and translation matrices are used to map the 3D coordinates to each camera coordinates [9], [10].

Frame Filtration– The processing system filters the set of all possible frames by eliminating the ones that do not capture any subject in the surveillance site and by selecting only the best camera frame among the sets of frames that capture exactly the same set of subjects. The selected frame is the one that achieves the maximum aggregate recognition probability of all subjects.

PTZ Camera Scheduling– The processing system carries out camera scheduling and frame assignment.

Recording and Tracking– The PTZ cameras apply the settings of camera planning and scheduling to capture the target subjects and start recording and tracking these subjects.

The total recognition probability function, $\psi(j)$, can be formulated as follows:

$$\psi(j) = \sum_{i=1}^{N_s} \gamma(i,j), \tag{1}$$

where N_s is number of subjects in frame j, and $\gamma(i,j)$ is the recognition probability of subject i captured in camera frame j. This equation sums the recognition probability value contributed by each covered subject i in frame j. The recognition probability $\gamma(i,j)$ of subject i captured in camera frame j can be formulated as follows:

$$\gamma(i,j) = W(i) \times R_{pose}(\theta_{tij}) \times R_{pose}(\theta_{pij}) \times R_{zoom}(z_{ij}) \times u_{cov}(i,j) \times u_{occ}(i,j), \tag{2}$$

where $W(i)$ is a weighting factor. The weight depends on the expected time for the subject to leave the site, and the overall recognition probability of the subject so far. R_{pose} and R_{zoom} are functions that determine the recognition probability based on the zoom level and the angle between subject and the optical axis of cameras, respectively. u_{cov} and u_{occ} can be used to assess the coverage and occlusion of the subject, respectively [6].

B. Proposed Clustering Approach

To populate frames efficiently, we utilize clustering as a pre-step for generating frames. The main advantage of the clustering process is enabling the system to focus on the

areas that are populated with more subjects. Grouping subjects into clusters can utilize one of the widely investigated clustering algorithms [11]. In this case, subjects are grouped based on many attributes, which can be translated eventually into distance. We define the cluster by three attributes: (i) cluster center, (ii) cluster direction, and (iii) cluster size. To determine whether a subject belongs to a cluster, we examine three parameters and check whether they are lower than pre-specified threshold values. These values are (1) the maximum allowed distance between the cluster center and the subject ($DIST\_TH$), (2) the maximum subject angle-offset from the cluster-center direction of motion ($ANGLE\_TH$), and (3) the maximum perpendicular-distance between the subject and the cluster direction of motion ($WIDTH\_TH$).

```
01. //Input: subjectList[], List of the active subjects
02. //Output: clusterList[],Cluster center list of the grouped subjects
03. Find-N-Clusters(){
04.   for (i = 0; i < subjectList.size(); i++){
05.     clusterList[i].Center = subjectList[i].Loc;
06.     clusterList[i].Dir = subjectList[i].Dir; // Direction angle
07.   }
08.   for (i = 0; i < clusterList.size(); i++){
09.     mergIndex = -1;
10.     for (j = 0; j < clusterList.size(); j++){
11.       Distance =
            DIST(clusterList[i].Center, clusterList[j].Center);
12.       // Calculate the perpendicular distance
13.       DistancePer =
            DISTPER(clusterList[i].Center, clusterList[j].Center);
14.       Direction = DIR(clusterList[i].Dir, clusterList[j].Dir);
15.       if (Direction > ANGLE_TH||Distance >
            DIST_TH||DistancePer > WIDTH_TH)
16.         continue;
17.       totalDist = Distance/DIST_TH +
            DistancePer/WIDTH_TH + Direction/ANGLE_TH;
18.       if (j == 0)
19.         minDist = totalDist;
20.       if (totalDist > minDist)
21.         continue;
22.       minDist = totalDist;
23.       mergIndex = j;
24.     } End of loop j
25.     if (mergIndex == -1)
26.       continue;
27.     clusterList[i] =
            Merge(clusterList[i], clusterList[mergIndex])
28.     clusterList[i].Center =
            Average(clusterList[i].Center,
            clusterList[mergIndex].Center)
29.     clusterList[i].Dir =
            Average(clusterList[i].Dir, clusterList[mergIndex].Dir)
30.     Remove(clusterList[j])
31.   } End of loop i
32. } // End of Find-N-Clusters()
```

Fig. 1. Simplified Algorithm to Find Cluster Centers and Subjects

K-Means is a popular clustering method, with N entities being mapped to K centers according to certain conditions quantified as a distance. The standard *Lloyad solution* to the K-Means problem involves two major steps: (1) assignment of nodes to clusters by finding the nearest cluster center to each node and (2) computing the new cluster center. In the considered clustering problem, the number and the locations of the K centers are unknown. Therefore, we use a variation of the *dendrogram hierarchical clustering* method by which each entity is initially considered as a cluster center. Subsequently, each pair of nodes with a minimum in-between distance are grouped to form one cluster. We then calculate the centers for the newly formed clusters and repeat the first step. Figure 1 explains the specific solution. Lines 11,13 and 14 calculate the Cartesian distance, the perpendicular distance, and the angle difference, respectively.

Figure 2 shows a simple illustration of cluster formulation. Cluster $R1$ is initially formed, then another subject at location A is added to form cluster $R2$. XA and XB represent the Cartesian distance, where XC and XD are the perpendicular distance between cluster $R1$ center and the subjects at location A and location B, respectively. Subjects at location F and location G cannot be added to the formed cluster. The directions of movement for these subjects are larger than the $ANGLE\_TH$. Subject at location B cannot be added too; because it has a perpendicular distance XD that is bigger than $WIDTH\_TH$. After finding the clusters, we start populating frames. We pick the centers of the clusters and identify the camera that can cover them with the best overall recognition probability. We also re-map the subjects in the provided clusters to form more precise clusters, without changing the cluster centers. Moreover, sub-clusters are also generated from the same centers. These sub-clusters are smaller than the maximum cluster size but have a higher resolution.

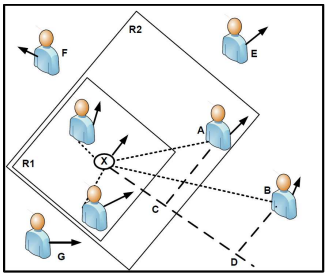


Fig. 2. Illustration of Cluster Formulation

After generating the frames using the clustering process, we apply the two camera scheduling algorithms in [5], specifically GBG and EBP. To distinguish the clustering versions from the original algorithms, we call the algorithms applying clustering *GBG with Clustering* (GBG-C) and *EBP with Clustering* (EBP-C), respectively. Furthermore, we introduce and apply a new enhancement, called *reachability enhancement*, to reduce the search domain in GBG-C and EBP-C. In particular, we can enhance the filtration by excluding all the frames that are unreachable due to a long time of computation, camera movement, and/or focus time. In EBP and EBP-C, we cannot determine the reachability beforehand as the location of the camera with respect to the frame is unknown until the computation of different plans is done.

IV. Performance and Evaluation

We have developed a simulator, called *AutoSurvSim*, in C++ for an AVS system, supporting the proposed solution. Without loss of generality, we assume a surveillance site with a rectangular area. Subjects arrive randomly from any of the four directions and enter the site with a random direction and a random speed. Poisson distribution is used to model subject arrivals, while a truncated Gaussian distribution is used for the speed, and a truncated uniform distribution is used for the direction [1]. The PTZ cameras are located at the perimeter of the site with different (x,y,z) locations. Moreover, a detailed characterization for each camera is implemented, including Pan-Tilt-Zoom speeds with maximum and minimum limits and a step size, refocusing speed, and sensor dimensions. Furthermore, we adopt a *frame independence policy* to exclude the effect of improving the recognition probability by capturing the same frames within a short period of time. Table I shows the main environment parameters and their values. The results

are collected by running the simulator on a system with 64-bit 3.6 GHz Quad-core CPU and 16 GB RAM. The simulations are run for the time required to process 50000 subjects.

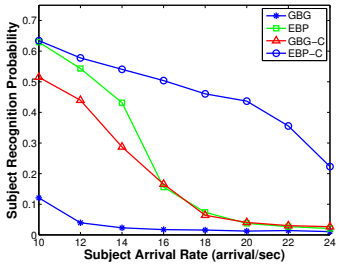
We consider two performance metrics: the *average subject recognition probability* and the *algorithm computation time*.

TABLE I
SUMMARY OF WORKLOAD CHARACTERISTICS [AVS, CLUSTERING]

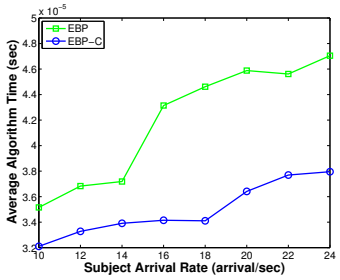
Parameter	**Model/Value(s)**
Scene Area, Cluster Width	80×60 m^2, 5.3m
Request Arrival Rate	Variable [10-24], Default = 14 Req./second
Request Speed	Truncated Normal Distribution [0.5-2.5]
Request Speed Rate	truncated Gaussian distribution [0.5-2.5]
Request Direction	Truncated Uniform Distribution [-40°40°] degree with $\perp$ to entrance side
PTZ Cameras, Zoom	8, Variable [1-50] levels
PTZ Pan, Tilt	Variable [0-180], [0-90] degrees
Sensor Resolution	$[1024 \times 768]$ pixels
Focus Time, T_p, T_r	0.5, 2, 2 seconds
$PTZStep$, Angle Threshold	4 steps, 25 degrees

V. RESULT PRESENTATION AND ANALYSIS

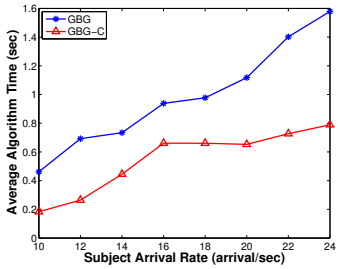
Figure 3(a) illustrates the effect of subject arrival rate on the average subject recognition probability. As the arrival rate increases, GBG and EBP degrade much faster than GBG-C and EBP-C, respectively. The clustering approach enables the system to become more scalable with respect to the arrival rate because the clustering process allows the search of only the areas that are populated with subjects. EBP-C behaves better at higher arrival rates. 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 3(b) and 3(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.



(a) Average Subject Recognition Probability



(b) Average Time



(c) Average Time

Fig. 3. Comparing Effectiveness of Clustering with Arrival Rate

VI. CONCLUSIONS

We have proposed a clustering-based approach for controlling the PTZ cameras in such a way that optimizes the overall subject recognition probability. The results demonstrate that the recognition probability and the required computation time are significantly improved by applying the proposed clustering approach as a pre-step in PTZ camera scheduling schemes.

REFERENCES

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

[2] H. El-Alfy, D. Jacobs, L. Davis, and L. Davis, "Assigning cameras to subjects in video surveillance systems." in *Proc. of International Conference on Robotics and Automation (ICRA)*, 2009, pp. 837–843.

[3] K.-Y. Li, J.-M. Liang, C.-S. Fan, K.-R. Wu, Y.-T. Lin, T.-Y. Lin, and Y.-C. Tseng, "A web-based, real-time video surveillance system by leveraging PTZ cameras," in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on*, April 2015, pp. 1–1.

[4] F. Rameau, C. Demonceaux, D. Sidib, and D. Fofi, "Control of a PTZ camera in a hybrid vision system," in *Computer Vision Theory and Applications (VISAPP), 2014 International Conference on*, vol. 3, Jan 2014, pp. 397–405.

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

[6] X. Zhou, R. T. Collins, T. Kanade, and P. Metes, "A master-slave system to acquire biometric imagery of humans at distance," in *Proc. of ACM SIGMM international workshop on Video surveillance*, ser. IWVS '03, 2003, pp. 113–120.

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

[8] S. P. Jeon, Y. S. Lee, and K. N. Choi, "Movement direction-based approaches for pedestrian detection in road scenes," in *Frontiers of Computer Vision (FCV), 2015 21st Korea-Japan Joint Workshop on*, Jan 2015, pp. 1–4.

[9] J. P. D. A. Forsyth, *Computer Vision: A Modern Approach*, 1st ed. Prentice Hall, 2002.

[10] M. Shah, "Fundamentals of computer vision." [Online]. Available: http://www.math.ucf.edu/csums/BOOK.PDF

[11] J. Kaur and H. Singh, "Performance evaluation of a novel hybrid clustering algorithm using birch and k-means," in *2015 Annual IEEE India Conference (INDICON)*, Dec 2015, pp. 1–6.