Sub-optimal Camera Selection in Practical Vision Networks through Shape Approximation

Huang Lee[1], Linda Tessens[2], Marleen Morbee[2],
Hamid Aghajan[1], and Wilfried Philips[2]

[1] Wireless Sensor Network Lab
Department of Electrical Engineering
Stanford University, Stanford, CA 94305
{huanglee,aghajan}@stanford.edu
[2] TELIN-IPI-IBBT
Ghent University
Sint-Pietersnieuwstraat 41, Ghent, Belgium
{marleen.morbee,linda.tessens}@telin.UGent.be

Abstract. Within a camera network, the contribution of a camera to the observations of a scene depends on its viewpoint and on the scene configuration. This is a dynamic property, as the scene content is subject to change over time. An automatic selection of a subset of cameras that significantly contributes to the desired observation of a scene can be of great value for the reduction of the amount of transmitted and stored image data. We propose a greedy algorithm for camera selection in practical vision networks where the selection decision has to be taken in real time. The selection criterion is based on the information from each camera sensor's observations of persons in a scene, and only low data rate information is required to be sent over wireless channels since the image frames are first locally processed by each sensor node before transmission. Experimental results show that the performance of the proposed greedy algorithm is close to the performance of the optimal selection algorithm. In addition, we propose communication protocols for such camera networks, and through experiments, we show the proposed protocols improve latency and observation frequency without deteriorating the performance.

1 Introduction

In many applications, the deployment of a camera network provides substantial advantages over a single fixed viewpoint camera. For example, in scene monitoring, camera networks can alleviate occlusion problems; in gesture recognition, cues coming from different viewpoints can lead to a more robust decision; in free viewpoint television, the quality of the rendered intermediate views benefits from a larger number of cameras.

Camera networks provide rich observation data for all kinds of applications, but observations among cameras are usually highly correlated when the cameras' views are overlapping, which results in redundant data during signal processing.

It is therefore beneficial, and from a practical point of view often necessary, to have a system that can fully exploit the information available in the network, while simultaneously keeping the redundancy under control. A possible way of achieving this is by selecting a limited number of cameras or views and transmitting and storing information only from them. Thus the amount of data that requires transmission, processing, and storage is greatly reduced and resources are saved. The recent introduction of "smart cameras" with on-board image processing and communication hardware allows for a distributed implementation of such a selection algorithm, hereby reducing the required communication bandwidth and spreading the computational burden. This is beneficial to the scalability of the system and can even allow the cameras to exchange information via wireless channels.

A selection algorithm for practical camera networks has to deal with computational, latency and bandwidth constraints. In [1], information-driven sensor querying and constrained anisotropic diffusion routing are proposed to select a subset of sensors from a general network. However, as shown in [2], in the case of camera networks, devising the necessary sensor and target models is far from straightforward. In [3], Yang *et al.* consider bandwidth and computational issues when tasking cameras in vision networks that determine the occupied area in an observed scene. In [4], the authors investigate the selection of cameras for the synthesis of intermediate views when observing a planar scene while considering the limited lifetime of battery-powered nodes in a wireless network. Also, a related topic is treated in [5], where real-time allocation of tasks in networks of smart cameras is studied.

As our interest lies in the observation of persons in a 3D scene, we focus on selecting a limited number of cameras from a network such that this subset constitutes the most complete view of the scene possible for the given number of selected cameras. In [6], we have proposed to use the occupancy area as a criterion for camera selection to provide an efficient representation of the observed scene. However, the proposed method involves a full search through all viable subsets, and is therefore not scalable to larger networks nor applicable at high frame rates. In this paper, we propose a greedy algorithm for camera selection and further discuss the practical issues to operate such a network including the network communication protocol.

The remainder of the paper is organized as follows: In Section 2 we describe system setup and assumptions. The algorithm is introduced in Section 3. Section 4 contains the performance evaluation and finally Section 5 concludes the paper.

2   System Setup

The system consists of multiple smart camera sensors that observe a room with one or more persons inside. The camera sensors are battery powered and can communicate with each other through wireless channels. The cameras' positions and orientations are fixed and calibrated. A base station is deployed to receive the observations from camera sensors and is responsible for coordinating all

sensors in the network. We assume that in the network there are N camera sensors denoted by C_i for $i = 1, \ldots, N$. The complete collection of cameras is the set $\mathbf{C} = \{C_1, \ldots, C_N\}$ where $|\mathbf{C}| = N$.

We assume that all cameras can exchange information with the base station. During the operations, each camera first processes the observed images locally, and then sends its processed information to the base station. The base station runs the camera selection algorithm based on the received data, and selects a subset of cameras denoted as $\mathbf{S} \subseteq \mathbf{C}$ where $|\mathbf{S}| = n \le N$. The selection result is then broadcasted to all the camera sensors, and only the selected cameras will later send their images to the base station. These images should comprise the most complete view of the scene possible for the given number of selected cameras. The subset $\mathbf{S}$ contains two types of cameras:

– The key camera: the camera with the view that contributes most to the desired observation of a scene at a certain time instant. The key camera is indicated by K.
– The helper cameras: cameras with views that complement the selected key view, such that the total selected view subset constitutes a significantly more efficient scene representation than the totality of the available views. The $n-1$ helper cameras are indicated by W_k where $k = 1, \ldots, n-1$.

Note that $\mathbf{S} = \{K\} \cup \{W_1, \ldots, W_{n-1}\}$, and the remaining $N - n$ cameras do not send any image data.

3   Camera Selection Algorithms

We discuss the details of the key camera and helper camera selection algorithms. The system diagram is depicted in Fig. 1.
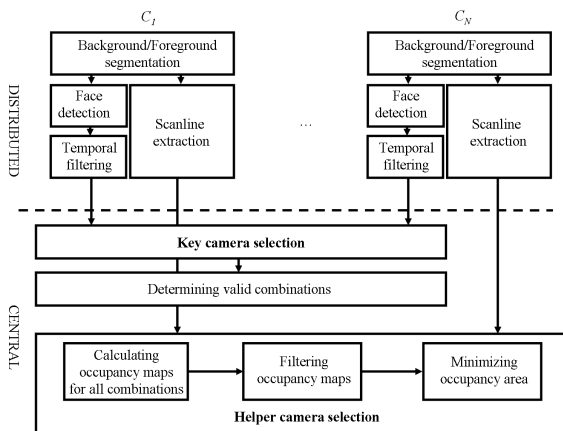

Fig. 1. Block diagram for camera selection

3.1   Key Camera Selection

To assign the role of key camera K in the camera network, we run the following algorithms on each of the smart cameras C_i in the network. The image captured by the i-th camera at a certain time instant t is denoted by $\mathbf{X}_i(t)$. In a first step, we segment the foreground $\mathbf{F}_i(t)$ and the background $\mathbf{B}_i(t)$ of the frames $\mathbf{X}_i(t)$ using the method of [7]. Then, we detect the frontal faces in the foreground regions of the frame with the object detector that was initially proposed by Viola *et al* [8] and then improved by Lienhart *et al* [9]. At each time instant t, the face detector returns the following values: $f_i(t)$ and $Q_i^l(t)$ $(l = 1, \ldots, f_i(t))$. $f_i(t)$ is the number of faces detected in the frame $\mathbf{X}_i(t)$. $Q_i^l(t)$ is a measure of the quality of the lth detected face. The lower this measure, the less certain the detection. In our implementation, we assume that the number of windows that have passed all classification stages and that constitute a detected face is such a measure.

   To deal with spurious face detections and to obtain smoothness over time, the decision on the key camera for time instant t not only depends on the current face detection output, but also on the previous observations. For each camera C_i, this temporal filtering is implemented as an exponentially weighted moving average (for $t \geq 2$):

$$Q_i^s(t) = \alpha \sum_{l=1}^{f_i(t)} Q_i^l(t) + (1 - \alpha) \sum_{l=1}^{f_i(t)} Q_i^s(t-1) \tag{1}$$

where $Q_i^s(t)$ is the smoothed face detection output of camera C_i at time instant t and α is a constant between 0 and 1 that determines the importance of previous observations. Then, the key camera at time instants $t \geq 2$ is

$$K(t) = \underset{C_i}{\operatorname{argmax}} \, Q_i^s(t) \tag{2}$$

In the remainder of this paper, we will leave out the time variable t, in order not to overload the notations.

3.2   Greedy Helper Camera Selection

We present a greedy algorithm that selects among the remaining $N - 1$ cameras those $w = n - 1$ helper cameras $\{W_1, \ldots, W_{n-1}\}$ that add most information to the image data coming from the already selected key camera.

   In [6], we showed that scan-lines can be effective information for camera selection. At each camera the scan-line is extracted from the silhouettes in the scene. This scan-line is the projection of the 2D-foreground silhouettes $\mathbf{F}_i$ to a 1D-line (see Fig. 2b). All the cameras send their (run-length coded) scan-lines to the base station. At the base station, the scan-line information from each camera is extended in a column-wise manner to a 2D image, such that we get a rough approximation of the original background B_i and foreground F_i extracted at the sensor nodes (see Fig. 2c). These approximations are denoted by $\mathbf{B}_{i,\text{sc}}$ and $\mathbf{F}_{i,\text{sc}}$.
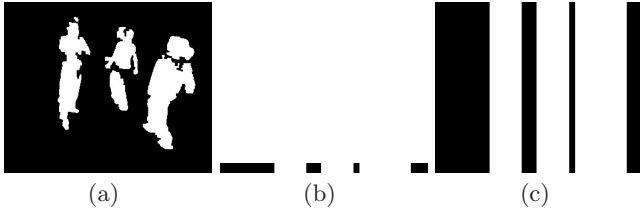
Fig. 2. Example of (a) Background/foreground segmentation ($\mathbf{B_i}$ and $\mathbf{F_i}$), (b) scan-line, and (c) column-wise extended scan-line ($\mathbf{B}_{i,\mathrm{sc}}$ and $\mathbf{F}_{i,\mathrm{sc}}$)

We can calculate the occupancy map, which is a 2D raster image, uniformly distributed in a plane P^2 horizontal to the ground floor of our observed 3D scene:

$$P^2 = [X_1, X_2) \times [Y_1, Y_2) \subset \mathbb{N}^2 \text{ and } Z = c. \tag{3}$$

The occupancy map is obtained by intersecting the visual hull reconstructed from the column-wise extended scan-lines $\mathbf{B}_{i,\mathrm{sc}}$ with the plane $Z = c$ [10].

We consider that the occupancy maps are a (very crude) shape approximations of the objects in the scene. The subset that yields the minimal occupied area is assumed to provide the most complete view on the scene. Therefore the size of the occupied area is considered the selection criterion in our algorithm and is denoted by $\mathbf{A(S)}$ where $\mathbf{S}$ implies this area is constructed based on the information $\mathbf{B}_{i,\mathrm{sc}}$ for $\forall C_i \in \mathbf{S}$.

Let us now assume that the currently activated cameras form the set $\mathbf{S}$. The algorithm then first combines the key camera along with all currently active cameras to form a new set $\mathbf{S'} = \mathbf{S} \cup \mathbf{K}$. During the selection processing, we assume the algorithm has to reevaluate the activity status of at least u cameras.

The proposed greedy selection algorithm has two parts. The first part of the algorithm greedily removes r cameras one at a time from the set $\mathbf{S'}$, and creates a new set $\mathbf{S''}$ which includes the remaining selected cameras. Given the number u, the number of cameras that will be first removed from the set $\mathbf{S'}$ is

$$r = \begin{cases} u, & if \ |\mathbf{S'}| \leq w, \\ \min\left(|\mathbf{S'}| - w + u, N\right), & if \ |\mathbf{S'}| > w. \end{cases} \tag{4}$$

The second part of algorithm then greedily adds $w - |\mathbf{S''}|$ cameras one at a time to $\mathbf{S''}$ if $w > |\mathbf{S''}|$, and we denote the final selection solution as $\hat{\mathbf{S}}$. The pseudo-code of the algorithms are summarized in Algorithm 1 and 2.

Initially, we can start the algorithm with all cameras activated, which means that $\mathbf{S}$ is equal to $\mathbf{C}$. The complexity of the optimal selection is $\binom{N-1}{n-1}$, and for the greedy algorithm, the complexity is reduced to $O\left(rN\right)$.
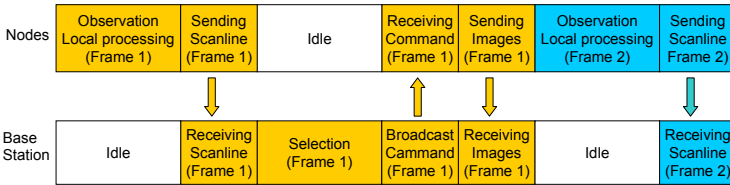
3.3   Operation Time Frame

The basic operation time frame is shown in Fig. 3 where different colors indicate operations on different image frames (i.e., captured at different time instances).

Algorithm 1. Greedy Selection Algorithm - Removing

Input: S′ (a set of currently selected cameras)
Output: S″ (a set of cameras after removing)

1: **for** $m = 1$ to r **do**
2:     $A_{min} \leftarrow \inf$
3:     **for** each camera C_i in **S′** **do**
4:         **if** $C_i \neq K$ **then**
5:             **S′** ← **S′**\\{C_i}
6:             $A \leftarrow \mathbf{A}(\mathbf{S'})$, the size of occupancy area given cameras **S′**
7:             **if** $A < A_{min}$ **then**
8:                 **S″** ← **S′**
9:                 $A_{min} \leftarrow A$
10:            **end if**
11:            **S′** ← **S′** ∪ {C_i}
12:        **end if**
13:    **end for**
14:    **S′** ← **S″**
15: **end for**

Nodes	Observation Local processing (Frame 1)	Sending Scanline (Frame 1)	Idle	Receiving Command (Frame 1)	Sending Images (Frame 1)	Observation Local processing (Frame 2)	Sending Scanline Frame 2)

Base Station	Idle	Receiving Scanline (Frame 1)	Selection (Frame 1)	Broadcast Cammand (Frame 1)	Receiving Images (Frame 1)	Idle	Receiving Scanline (Frame 2)

Fig. 3. The time frame of the basic operation scheme

The sensor nodes first make observations and process the images locally. The main operations this processing encompasses are background subtraction and face detection. Then, each node sends its scan-line to the base station. After receiving the scan-line from all nodes, the base station runs the greedy selection algorithm and broadcasts the result. Finally, the selected nodes transmit their images to the base station, after which the nodes start making new observations for the next frame and a new cycle starts.

From Fig. 3, it can be observed that both the base station and camera nodes have idle time slots, which increases the interval between observations. In order to increase the observation frequency, we propose an interleaving scheme as shown in Fig. 4. In this scheme, the operations on different image frames are interleaved to minimize the idle time. While the nodes are making observations for Frame 1 (marked by orange in Fig. 4), the base station decides on the camera selection based on the observations of a previous frame (Frame 0, marked by yellow). After the selection is completed, each node receives the broadcast from the base station and the selected ones transmit their image frames (Frame 0), and once

---

Algorithm 2. Greedy Selection Algorithm - Adding

Input: $\mathbf{S}''$ (a set of cameras after removing)
Output: $\hat{\mathbf{S}}$ (a set of new selected cameras)

```
 1: Ŝ ← S″
 2: for m = 1 to w − |S″| do
 3:     A_min ← inf
 4:     for each camera C_i in C do
 5:         if n ∉ S″ then
 6:             S″ ← S″ ∪ {C_i}
 7:             A ← A(S″), the size of occupancy area given cameras S″
 8:             if A < A_min then
 9:                 Ŝ ← S″
10:                 A_min ← A
11:             end if
12:             S″ ← S″\{C_i}
13:         end if
14:     end for
15:     S″ ← Ŝ
16: end for
```
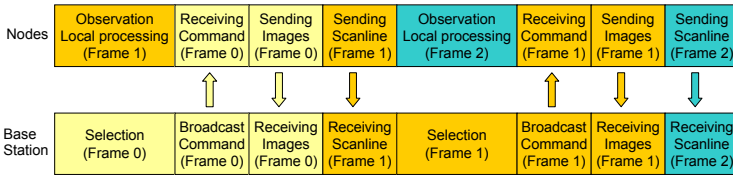

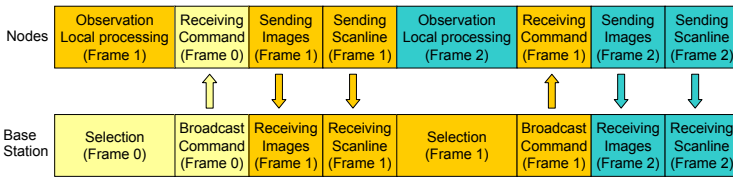**Fig. 4.** The time frame of the interleaving operation scheme

Fig. 5. The time frame of the advance operation scheme

the image frames (Frame 0) are sent, each node starts sending the scan-line of frame (Frame 1).

Although the interleaving operation scheme improves the observation frequency, it increases the delay between the observation of a frame and the same frame received at the base station. To decrease this delay, we propose the advance operation scheme shown in Fig. 5. In this scheme, the camera nodes receive selection results from Frame 0 (marked by yellow in Fig. 5) right after making the observations for frame (Frame 1, marked by orange). Instead of sending the

image frames (Frame 0) as in the interleaving scheme, the selected nodes now transmit image frame (Frame 1). In other words, we assume that the difference between successive observations is small. Under this assumption, the base station can select the current camera set based on the previous observation. This scheme is useful when the scene changes in the room are not fast, resulting in similar selection results between successive observations.

4   Performance Evaluation

We evaluated the performance of the proposed greedy camera selection algorithm in Section 3.2 by comparing it to its full search counterpart, and assessed how the application of the advance operation scheme proposed in Section 3.3 influences the accuracy of the system.

Experimental data to test the methods on was recorded using the whole set $\mathbf{C}$ of $N = 10$ cameras. Five of these were Logitech QuickCam Pro 5000 cameras and the five others were Logitech QuickCam Sphere MP. The cameras were calibrated using the method for multi-camera self calibration [11]. Sequences were recorded at 5 frames per second and at a resolution of 352×288. Only the starting points of the recordings were synchronized. The parameters of the background/foreground segmentation and the face detection are summarized in Table 1. We allowed the background/foreground segmentation algorithm to build up its background model during 30 frames at the start of each sequence. These first 30 frames of each sequence are not considered in the experiments in this section. The voxel volume for visual hull reconstruction was $[0, 200) \times [0, 100) \times [0, 50) \subset \mathbb{N}^3$ where each voxel is a cube with edges of 0.04 meter. The plane P^2 is the plane in the voxel volume at $Z = 1.29$ meter. Four scenarios were considered in our experiments, with number of people from one to four.

Table 1. Parameters for background/foreground segmentation and face detection

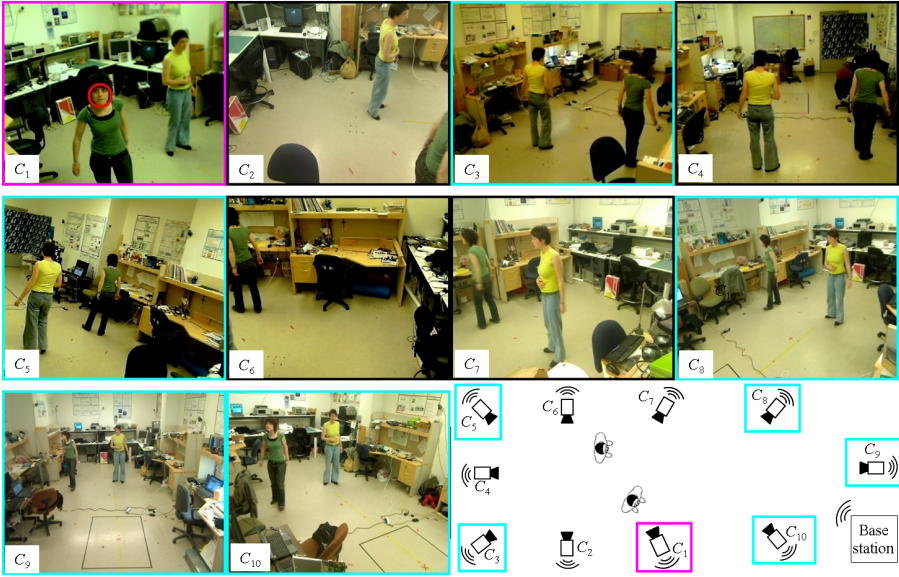| Parameters Background/Foreground Segmentation			
L (color comp.)	128	L (color co-occ.)	64
N_1 (color comp.)	15	N_1 (color co-occ.)	25
N_2 (color comp.)	25	N_2 (color co-occ.)	40
α_1	0.1	α_2	0.005
α_3	0.1	T	0.9
δ	2	MINAREA	15.0
UPDATE_TRESH	0.5		
Parameters Face Detection			
scale factor		1.10	
min. number (-1) of neighbors		2	
min. window size		5×5	
classifier training window size		20×20	

Fig. 6. Example of the selection of 6 out of 10 cameras for a 2-person scene. The views of the 10 cameras $(C_1, \ldots, C_{10})$ are shown. In the bottom-right corner, we depicted a top view of the scene which shows its geometry and the positions of the cameras and persons. The selected key camera is marked by a magenta bounding box. The helper cameras are marked by the cyan bounding boxs.

Fig. 6 shows a visual example of the selection of $n = 6$ cameras from 10 for a 2-person scene. We display the views of all the cameras $C_1, \ldots, C_{10}$. To give an insight into the system setup, we depicted in the bottom-right corner a top view of the scene, which indicates the relative positions of the ten cameras and the persons in the scene. The selected key camera is marked by a magenta bounding box. The detected face is indicated by a red circle. Due to the current and previous face detections, this camera was chosen to be the key camera (according to Eq. (2)). The helper cameras are marked by a cyan bounding box. We can observe from the displayed views that the selected subset gives us a complete view of the persons, and that the non-selected cameras add redundant information.

4.1 Accuracy Evaluation

We evaluated the accuracy of the proposed greedy algorithm. We first reconstructed the 3D visual hull $\mathbf{H}(\hat{\mathbf{S}})$ based on the foregrounds $\mathbf{F}_i$ from the cameras in the greedy solution set $\hat{\mathbf{S}}$ such that $\forall C_i \in \hat{\mathbf{S}}$. We also reconstructed the 3D visual hull $\mathbf{H}(\mathbf{C})$ based on the foreground $\mathbf{F}_i$ from all cameras $C_i \in \mathbf{C}$. The visual hull $\mathbf{H}(\mathbf{C})$ is considered the correct shape of the objects, and is served as a performance baseline. Finally, we reconstructed the 3D visual hull $\mathbf{H}(\hat{\mathbf{S}}_{opt})$ based on the foreground $\mathbf{F}_i$ from the cameras in the optimal solution set $\hat{\mathbf{S}}_{opt}$,

Table 2. Mean voxel difference for the optimal and greedy selection methods for four different scenarios. The average voxel volume of $\mathbf{H}(\mathbf{C})$ is shown in the third column.

Scenario	# frames	average voxel volume	$\hat{d}^{\text{optimal}}$	$\hat{d}^{\text{greedy}}$
1 person	1527	653.66	786.42	748.21
2 persons	2100	2530.20	830.03	859.85
3 persons	789	4799.33	1887.43	1792.36
4 persons	266	8806.56	1754.52	1639.69

where the optimal solution selects $n - 1$ helper cameras which give the minimal occupancy area.

Given the reconstructed visual hulls at each time instance, we calculated the number of different voxels, denoted $d^{\text{greedy}}(\mathbf{H}(\hat{\mathbf{S}}), \mathbf{H}(\mathbf{C}))$, which are the differences between the greedy solution visual hull $\mathbf{H}(\hat{\mathbf{S}})$ and the benchmark visual hull $\mathbf{H}(\mathbf{C})$. For the optimal solution, we also calculated the number of different voxels and denote it by $d^{\text{optimal}}(\mathbf{H}(\hat{\mathbf{S}}_{opt}), \mathbf{H}(\mathbf{C}))$.

In Table 2, we compare for the greedy and optimal methods the mean value of the number of different voxels, denoted by $\hat{d}^{\text{greedy}}$ and $\hat{d}^{\text{optimal}}$, over all frames of the sequences with a certain scenario. The lower this number, the higher the quality of the observation with the selected camera subset. The number of frames available per scenario is indicated in the second column, and the average voxel volume of $\mathbf{H}(\mathbf{C})$ in the third column. In these experiments, $n = 6$ cameras were selected among 10 cameras, and in each time frame at least $u = 2$ cameras were updated.

We observe that the optimal and greedy methods yield similar results. In some cases, the greedy method even outperforms the optimal one, despite its smaller computational complexity. This is possible because the occupied area is only an approximation of the shape of the people present in the scene. The subset of cameras that minimizes the occupancy area does not necessarily lead to the solution that gives the best visual hull.

4.2   Reduction of Delay

When the advance operation scheme (Fig. 5) is applied, the selection decision at time instant t is based on observations of the previous time instant $t - 1$. In order to evaluate the impact of this shift on the accuracy, we process the observations in a similar way as in the previous section. Only now, at time instant t the foreground silhouettes $\mathbf{F}_i$ from which the visual hull was reconstructed for accuracy evaluation were selected based on the selection of the previous time instant $t - 1$.

The experimental results are shown in Table 3 where $n = 6$ cameras were selected among 10 cameras, and in each time frame at least $u = 2$ cameras were updated. Comparing Tables 2 and 3, the accuracy is comparable in both cases. Therefore, the introduced delay has almost no impact on the performance.

Table 3. Mean voxel difference for the optimal and greedy selection methods with one frame delay. The average voxel volume of $\mathbf{H}(\mathbf{C})$ is shown in the third column.

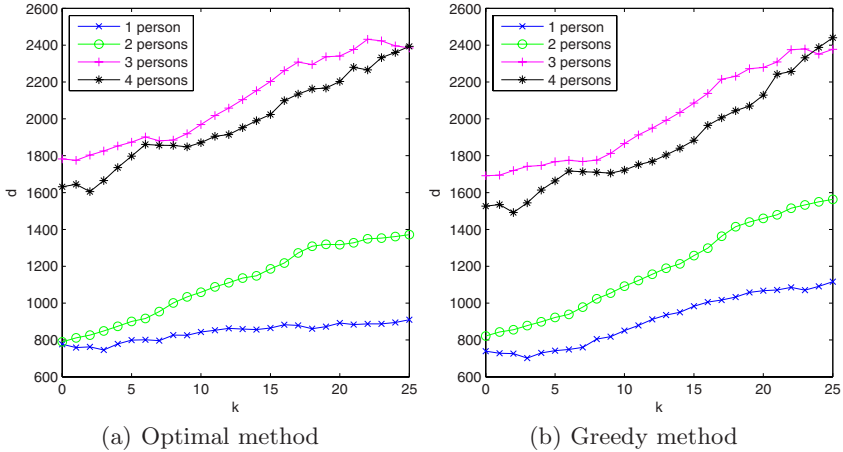| Scenario	# frames	average voxel volume	$\hat{d}^{\text{optimal}}$	$\hat{d}^{\text{greedy}}$
1 person	1517	654.12	759.24	727.67
2 persons	2090	2533.76	811.79	842.98
3 persons	785	4811.50	1775.05	1694.33
4 persons	265	8822.06	1644.16	1534.66

(a) Optimal method   (b) Greedy method

Fig. 7. Mean voxel difference d for the optimal and greedy selection methods for four different scenarios as a function of the delay k between the observation and the selection decision

We also investigated the delay impact on the performance when the delay k between observation and selection decision is more than one frame. The performance over different delays k is plotted in Fig. 7. The figure shows that a delay has a similar impact on the optimal and the greedy methods. It can also be observed that except for the four persons scenario, delays of up to 5 frames result in only a minor drop in quality. Therefore, we can further reduce the data transmission by transmitting the scan-lines every k frames instead of all frames. In other words, we can use the same selection results for every k frames for small k.

5   Conclusions

A greedy camera selection algorithm was proposed for real time network operation. The selection is based on the scan-line information that requires low data rate transmission, and therefore the algorithm is suitable for wireless networks. We used 3D shape reconstruction to compare the proposed greedy algorithm and optimal selection algorithm. Experimental results showed that the proposed algorithm provides a performance very close to the optimal results. Also, two different

network operation protocols were proposed. The first scheme aims to improve the sensor observation frequency and the second scheme improves the delay latency between view observation and image transmission. Experimental results also verified that the proposed protocols improve observation frequency and latency without degrading much the performance of the 3D shape reconstruction.

Acknowledgement

References

1. Chu, M., Haussecker, H., Zhao, F.: Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks. International Journal of High Performance Computing Applications 16, 293–313 (2002)
2. Soro, S., Heinzelman, W.B.: On the coverage problem in video-based wireless sensor networks. In: 2nd International Conference on Broadband Networks, BROADNETS 2005, Boston, MA, United States, vol. 2005, pp. 9–16 (2005)
3. Yang, D., Shin, J., Ercan, A.O., Guibas, L.: Sensor tasking for occupancy reasoning in a camera network. In: Proceedings of IEEE/ICST 1st Workshop on Broadband Advanced Sensor Networks (BASENETS 2004) (2004)
4. Yu, C., Soro, S., Sharma, G., Heinzelman, W.: Lifetime-distortion trade-off in image sensor networks. In: 2007 International Conference on Image Processing, Proceedings, vol. V, pp. 129–132 (2007)
5. Bramberger, M., Rinner, B., Schwabach, H.: A method for dynamic allocation of tasks in clusters of embedded smart cameras. In: Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, Waikoloa, HI, United States, vol. 3, pp. 2595–2600 (2005)
6. Morbee, M., Tessens, L., Lee, H., Philips, W., Aghajan, H.: Optimal camera selection in vision networks through shape approximation. In: 2008 International Workshop on Multimedia Signal Processing (October 2008)
7. Li, L., Huang, W., Gu, I.Y.H., Tian, Q.: Foreground object detection from videos containing complex background. In: MULTIMEDIA 2003: Proceedings of the eleventh ACM international conference on Multimedia, pp. 2–10. ACM, New York (2003)
8. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features (2001)
9. Lienhart, R., Maydt, J.: An extended set of haar-like features for rapid object detection. In: International Conference on Image Processing 2002. Proceedings, vol. 1, pp. I–900–I–903 (2002)
10. Hoover, A., Olsen, B.D.: A real-time occupancy map from multiple video streams. In: International Conference on Robotics and Automation (ICRA), pp. 2261–2266 (1999)
11. Svoboda, T., Martinec, D., Pajdla, T.: A convenient multi-camera self-calibration for virtual environments. Teleoperators and Virtual Environments 14, 407–422 (2005)