A POMDP-based Camera Selection Method

Li Qian[1,2], Sun Zheng-Xing[1], and Chen Song-Le[1]
[1]*State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China*
[2]*Institute of Meteorology, PLA University of Science and Technology, Nanjing 211101, China*

Keywords: Camera Selection, POMDP, Video Analysis, Multi-camera System.

Abstract: This paper addresses the problem of camera selection in multi-camera systems and proposes a novel selection method based on a partially observable Markov decision process model (POMDP). And an innovative evaluation function identifies the most informative of several multi-view video streams by extracting and scoring features related to global motion, attributes of moving objects, and special events such as the appearance of new objects. The experiments show that these proposed visual evaluation criteria successfully measure changes in scenes and our camera selection method effectively reduces camera switching.

1 INTRODUCTION

Multiple cameras are widely used for security surveillance, human–computer interaction, navigation, and positioning. However, they introduce issues related to deployment and control the cameras, real-time fusion of video streams with high resolution and high frame rates, and selection and coordination of the cameras (Soro and Heinzelman, 2009). Camera selection, which involves selection of one or more cameras from a group of cameras to extract essential information, is a particularly challenging task in multi-camera systems.

A number of previous studies have investigated the issues related to camera selection. Li and Bhanu (Li and Bhanu, 2009) proposed a game-theoretic approach to hand-off the camera with the global utility, camera utility, and person utility determined by user-supplied criteria such as the size, position, and view of the individual being tracked. Daniyal, Taj, and Cavallaro (Daniyal et al., 2010) proposed a Dynamic Bayesian Network approach that uses object- and frame-level features. Bimbo and Pernici (Bimbo and Pernici, 2006) selected optimal parameters for the active camera on the basis of the appearance of objects and predicted motions to solve the traveling salesman problem. Tessens et. al (Tessens et al., 2008) used face detection and the calculated spatial position of the target to select a primary view and a number of additional views. All methods mentioned above use low-level image features to evaluate visual information; high-level information in video streams, such as local salient movement details and specific events, is more informative.

In this paper, we present a novel method for camera selection based on a partially observable Markov decision model (POMDP) and use the belief states of the model to represent noisy visual information. By considering current states and anticipated transition trends with the cost generated by camera switching, the visual jitters that arise from frequent switching can be effectively reduced. Our evaluation function are presented for visual information, which is designed to reflect the richness of information in each view by extracting global motion, properties of moving objects in the scene, and specific events.

2 DYNAMIC CAMERA SELECTION BASED ON A POMDP MODEL

The camera selection problem can be described as follows. A multi-camera system has N cameras $(C^1, C^2, ...C^N)$ with partially or completely overlapped FOVs, and one node is designated as the central controller for scheduling according to a selection policy that is computed offline. In our

Qian L., Zheng-Xing S. and Song-Le C..
A POMDP-based Camera Selection Method .
DOI: 10.5220/0004228907460751
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2013), pages 746-751
ISBN: 978-989-8565-47-1

method, the central controller selects only one camera C^* online as the optimal camera at fixed time intervals of duration Δt. At each time step t, visual features indicating global motion, properties of objects, and specific events in the view of each camera are extracted and scored (see Section 3). The scores are then sent to the central controller, which makes dynamic selection decisions based on current and previous camera view scores. Although the camera selection problem can usually be modeled as a finite-state Markov decision process (MDP), when an observed state contains errors caused by factors such as illumination, occlusion, and camera shock and does not reflect the actual state, we must implement sequential decision making based on partially observable states. Owing to uncertainty, we model this dynamic process as a POMDP, which is an extended version of an MDP.

2.1 Definition of POMDPs

A POMDP can be formally defined as a 6-tuple $< \mathcal{S}, \mathcal{A}, \Omega, \mathcal{T}, O, \mathcal{R} >$, where $\mathcal{S}$ is a finite set of all possible underlying states, $\mathcal{A}$ is a finite set of actions, i.e., available control choices at each time instant, Ω is a finite set of all possible observations that the process can provide, $\mathcal{T}$ is a state transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ that encodes the uncertainty about the evolution of the states of the process, O is an observation function $O : \mathcal{S} \times \mathcal{A} \to \Omega$ that relates the process outputs (camera observations) to the true underlying state of the process, and $\mathcal{R}$ is an immediate reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbf{R}$ that assigns real-valued rewards to the actions that may be performed in each of the underlying process states.

2.2 Selection Policy

On the basis of our description of the camera selection problem, we formulate the POMDP as follows.

1) *System state vector*

The system state vector consists of the currently selected results and visual information scores. At time step t, the system state is represented as $S_t = [c_t^i, s_t^1, s_t^2, ..., s_t^N]$, where c_t^i is the best camera i that is selected at time t, and $s_t^k, k \in \{1, 2, ..., N\}$ is the actual visual information score for camera k. And the score values s_t^k are uniformly discretized with m quantization levels and normalized to [0,1] to produce the range $s_t^k \in \{0, 1, ..., m-1\}$.

2) *Actions*

An action is a vector represented as $a_t = [a_t^1, ..., a_t^N]$, where at time step t if the i th camera is selected, $a_t^i = 1$; else $a_t^i = 0$.

3) *Observation state*

The observation state is a collection of observations from all cameras and is defined as the vector $O_t = [c'^i_t, o_t^1, o_t^2, ..., o_t^N]$. c'^i_t is the camera i that is selected at time t. Because this component has no error, it should be the same as the component c_t^i of the system state, which is similar to the definition of system states. Each observation $o_t^k, k \in \{1, 2, ..., N\}$ is the visual score computed by our method for camera k at time step t obtained by extracting visual features from the video stream and scoring them. The number of observations is equal to the number of system states.

4) *State transitions*

The state transition probability $p_{s'|as}$ describes the differences in the scene between the views taken in adjacent time steps. Because the selection action does not affect the visual measures and state transitions $p_{s'|as} = p_{s'|s}$, the state transition will be based on the visual score component s_t^k. We assume that the visual scores for the cameras are independent, i.e.,

$$\begin{aligned} &\Pr(S' = (s_{t+1}^1, s_{t+1}^2, ..., s_{t+1}^N) \mid S = (s_t^1, s_t^2, ..., s_t^N)) \\ &= \Pr(s_{t+1}^1 | s_t^1) \cdot \Pr(s_{t+1}^2 | s_t^2) \cdot ... \cdot \Pr(s_{t+1}^N | s_t^N) \end{aligned} \quad (1)$$

For each discrete state component $s_t^k \in \{0, 1, ...m-1\}$ probability of transition to a neighboring state is higher than that to more distant states. Thus, we set the transition probabilities on the basis of distances between states as follows:

$$\Pr(s_{t+1}^i = u \mid s_t^i = v) = \begin{cases} \frac{1}{m-1}\left(1 - \frac{(u-v)^2}{\sum_{r=0}^{m-1}(r-v)^2}\right) & u, v \in \{0, 1, ..., m-1\} \\ 0 & u \notin \{0, 1, ..., m-1\} \quad or \quad v \notin \{0, 1, ..., m-1\} \end{cases} \quad (2)$$

This equation defines the transition probabilities between all states in the state space.

5) *Observation function*

The observation function $p_{o'|as'}$ indicates the likelihood of the observation state being o' if the system state s' performs action a. Because the selection actions do not affect the camera observations or the computed scores, we set

$p_{o'|as'} = p_{o'|s'}$. Also, we assume that the states in the observation state space for different cameras are independent. The observation probability is defined as follows:

$$\Pr(o_t^i{=}u \mid s_t^i = v) = \begin{cases} \frac{1}{m-1}\left(1-\frac{(u-v)^2}{\sum_{r=0}^{m-1}(r-v)^2}\right) & u,v \in \{0,1,...,m-1\} \\ 0 \quad u \notin \{0,1,...,m-1\} \quad or \quad v \notin \{0,1,...,m-1\} \end{cases} \tag{3}$$

6) *Immediate rewards*

For each action, we define an immediate reward or cost to measure the degree of optimization that would result from that action. We use the visual score from each camera as a positive reward and use the camera switching cost $c_{\cos t}$, which represents the visual jitter caused by frequent switching as a negative reward. Therefore, the immediate reward after camera selection is defined as

$$R(S_t, a_t^i) = \tau s_t^i + (1-\tau) c_{\cos t}\, \delta(a_{t-1}^i), \tag{4}$$

where $\tau \in [0,1]$ is the weight coefficient between the positive and negative rewards, and the δ function $\delta(a_{t-1}^i) = 1$ if the camera selected at time step t was also selected at time step $t-1$; else $\delta(a_{t-1}^i) = 0$.

Given the belief state $b(s)$ for the camera system state s at time step t, the scheduling agent attempts to maximize the total reward $V^{\pi^*}(b(s))$ by selecting the best camera a_t^k on the basis of the optimal policy π^*. This condition is represented as follows:

$$V_t^{\pi^*}(b(s)) = \max_{a_t^k \in A}\{R(b(s), a_t^k) + \gamma \sum_{s' \in S} p(s, a_t^k, s') V_{t+1}^{\pi^*}(b(s'))\}, \tag{5}$$

where $\gamma \in [0,1]$ is a discount factor that controls the future impact of rewards so that the effect of a reward decays exponentially with respect to elapsed time. If o is the observation after action a has been executed, the next belief state $b_a^o(s')$ is calculated on the basis of Bayesian theory as follows:

$$b_a^o(s') = \frac{p_{o|s'a}}{p_{o|ab}} \sum_{s \in S} p(s' \mid s, a) b(s), \tag{6}$$

where $p_{o|ab}$ is a normalized constant defined as $p_{o|ab} = \sum_{s' \in S} p_{o|s'a} \sum_{s \in S} p_{s'|sa} b(s)$.

Formula (5) can usually be computed iteratively using dynamic programming; the computational complexity increases exponentially with respect to the scale of the problem. Therefore, a direct solution to our POMDP is unfeasible because the problem is intractable, and thus, we use the Perseus method (Spaan and Vlassis, 2005), which is a point-based approximation. We sample randomized a number of belief state points b as a belief set and compute the reward values for this belief set. We save the results as a set of value vectors $\{\alpha_1, \alpha_2, ..., \alpha_n\}$ in which each vector α_i is associated with a selected action a_i so that the vector contains the same number of components as the state space. When the central controller selects the best camera online, it transforms the observed states into belief states and makes its decision on the basis of the following relation:

$$a = \arg\max_{a_i \in A}(b(s') \bullet \alpha_i), \tag{7}$$

The value vector α_i that has its inner product with the current belief state at maximum is selected, and the corresponding action a_i is selected as the best camera.

3 VISUAL INFORMATION MEASURE

In this section, we propose a measure that evaluates the quality of the image captured by a camera by extracting features indicating motion, properties of objects, and special events and expressing features as a motion score S_m^i, an object score S_{obs}^i, and an event score S_e^i, respectively. The final visual score as then calculated as follows:

$$S^i = w_1 S_m^i + w_2 S_{obs}^i + w_3 S_e^i, \tag{8}$$

where w_1, w_2, and w_3 are weight coefficients such that $\sum_{i=1}^{3} w_i = 1$. To simplify the exposition that follows, we denote the motion scores, object scores, and event scores without a superscript for the camera as S_m, S_{obs}, and S_e, respectively.

3.1 Global Motion

The degree of motion in a video stream reflects the ability of the camera to capture real world changes. We adopt the method presented in (Zhang et al., 2011) for detecting moving objects and mitigating the effects of illumination and shadows. Then we determine the degree of global motion in a video frame by calculating the ratio of the foreground area

to the area of the entire image. We assume that the ratio will increase significantly when a new object enters the scene or objects are close to the camera. However, the ratio exceeding a certain threshold implies that noise has been introduced by factors such as illumination. Thus, we score the global motion contained in a binary motion image as follows:

$$S_m = \begin{cases} \frac{1}{\lambda} r & r \le \lambda \\ \frac{1-r}{1-\lambda} & r > \lambda, \end{cases} \quad (9)$$

where r is the ratio of the area of the moving object to the area of the entire image, and λ is the best ratio required by applications.

3.2 Object Properties

Individuals and special objects are the most attractive elements in video surveillance applications. To properly measure the properties of objects in video streams, we focus on motion saliency for individual objects and the degree to which the objects in a scene occlude one another. To detect the objects in a scene and appropriately separate objects that are partially overlapped, we use the method proposed by Hu (Hu et al., 2006) for extracting bounding boxes for objects even when they overlap. We then use the bounding boxes to assign scores on the basis of local motion saliency and occlusion between two overlapped objects. Finally, we calculate an overall score for the properties of the object in this scene as a weighted sum of the two scores:

$$S_{obs} = \beta \frac{1}{|OBS|} \sum_{k \in OBS} S_l^k + (1-\beta) S_{oc}, \quad (10)$$

where S_l^k is the local motion saliency score for object k, $|OBS|$ is the number of objects in the scene, S_{oc} is the occlusion score, and β is a weight factor.

1) *Local motion*

We use the Harris 3D spacetime interest point method (Laptev, 2005) to detect salient motion details in the bounding boxes of objects and set the saliency measure S_l^k of the k th object to be the normalized number of 3D interest points detected in the bounding box. Although the interest point method is computationally expensive, we can control the computation process by limiting the search to a small region. Our experiments show that this method can be effectively applied in real-time video surveillance applications.

2) *Object occlusion*

One of the most serious issues in video surveillance, tracking, and other applications is occlusion between objects in a scene. In multi-camera systems, selecting a camera with the least occlusion is an effective solution. For this purpose, we measure the degree of occlusion between the objects in a scene on the basis of intersection of the bounding boxes for different targets and use this measure to assign a score that reflects the occlusion. The larger the areas with intersections, the lower the score. Therefore, we define the occlusion score as follows:

$$S_{oc} = \begin{cases} 1 & |OBS| \le 1 \\ 1 - \sum_{i=1}^{|OBS|} \sum_{j=i+1}^{|OBS|} \frac{\#(Rc_i \cap Rc_j)}{\min(\#Rc_i, \#Rc_j)} & |OBS| > 1 \end{cases}, \quad (11)$$

where Rc_i denotes the bounding box of the i th object and $\#$ denotes the area of the box.

3.3 Event Detection

It is necessary to detect events of interest in videos. In this paper, we focus on the entrance of new objects in a video Frame. We determine an entrance region in the image called the "inner region" either by predefining it or through training and monitoring the ratio r_1 of the area of motion in the inner region to the entire area of the inner region. When r_1 exceeding the threshold Th_1, it indicates that an object possibly enters the scene. To avoid false detection with the ratio r_1 due to the movement of individuals present in the scene, we introduce an external region around the entrance called the "outer region" and determine the ratio r_2 of the area of motion in the outer region to the area of the entire outer region. When r_2 exceeds a threshold Th_2, motion in the entrance is assumed to be the motion of an object that is already in the scene. Also, an entrance event is related to time. Therefore, we set

$$S_e = \begin{cases} e^{-\frac{T}{\sigma^2}} & r_1 > Th_1 \ \ and \ r_2 < Th_2, \\ 0 & otherwise, \end{cases} \quad (12)$$

where T is the time that has elapsed since the condition for a new object entering the scene is satisfied, so that the event score is gradually lowered as time elapses.

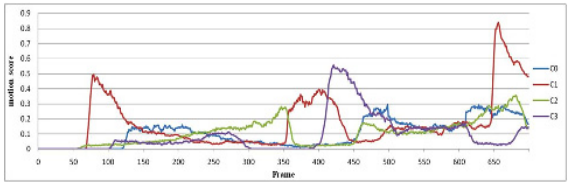
4 EXPERIMENTS AND RESULTS

4.1 Experimental Setup

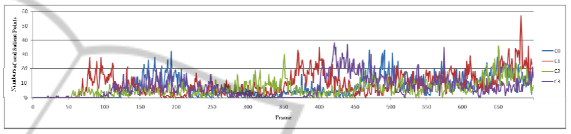
We conducted experiments on a personal computer to simulate the process of camera selection. We generated optimal policies offline using the Perseus' point-based method implemented using C++ and quantified the camera scores on an eight-point scale (m=8) and saved the optimal policies as value vectors. We used publicly available datasets composed of sequences from POM (Fleuret et al., 2008) and HUMANEVA (Leonid et al., 2010) to evaluate the performance of our method. The POM dataset use four cameras, while the HUMANEVA dataset uses seven cameras.

4.2 Visual Information Measurement Analysis

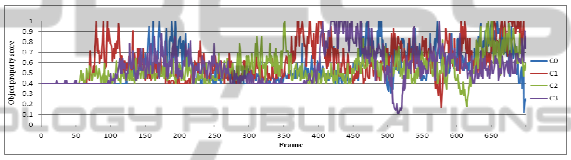
In this section, we evaluate visual information scores obtained from our method for the POM Terrace1 sequence. We denote the images from the four cameras as C0, C1, C2, and C3. In the scene captured by frames 1 through 700 of the Terrace1 sequence, no individual is present in the scene, thereafter, one person enters the scene, followed by two people with no occlusion, two people occluding each other, and eventually three people are present in the scene. After several experiments and analysis, we set the parameters λ , β , w_1, w_2 , and w_3 respectively. Therefore the global motion score curve with $\lambda = 0.6$ shown in Figure 1(a) reflects the ability of the camera to capture the motion in the observed scene, as well as the number of objects and their distance from the camera. The numbers of salient points for different views are shown in Figure 1(b). The object property scores with $\beta = 0.6$ in Figure 1(c) indicate that cameras can detect the same significant object properties, when global motion plays a dominant role in the videos. Also, when cameras provide similar global information, the local motion saliencies are different owing to different camera orientations and relative directions of motion of the objects from these perspectives, for example, the scores for frames 160 to 200 in Figure 1(c) and the occlusion between objects is appropriately detected. The curves in Figure 1(d) show the scores for measuring entrance events that appropriately indicate that two people have entered the scene during this period. Finally, Figure 1(e) displays the overall visual information score curves with the weights w_1 , w_2 , and w_3 set to 0.5, 0.2, and 0.3, respectively. Thus, the evaluation of visual information by our method in the analyzed video streams can appropriately reflect changes in the scene and details and special events of interest to observers.



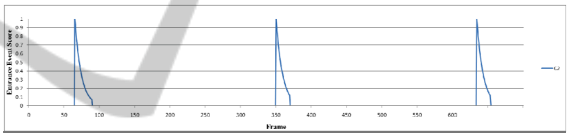
(a) Global motion scores



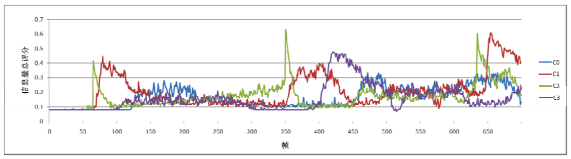
(b) Numbers of local salient points



(c) Object property scores, $\beta = 0.6$



(d) Entrance event scores for C2



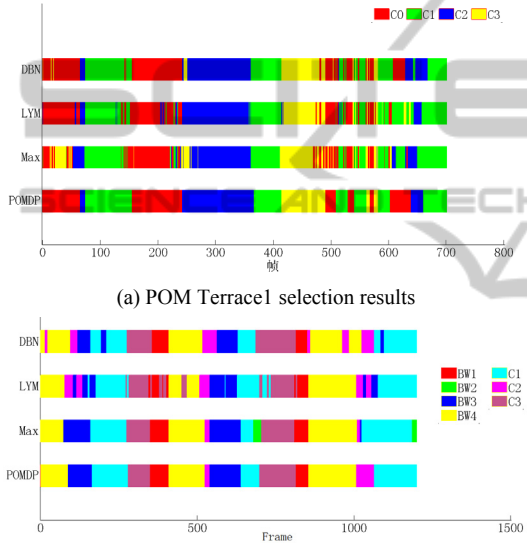
(e) Visual information scores

Figure 1: Visual information measurement results.

4.3 Selection Results and Analysis

We compared our method of camera selection (POMDP) to the state-of-the-art camera selection methods based on greedy criteria for maximum visual scores (Max), game theory 0 (LYM), and Dynamic Bayesian Networks 0 (DBN). Our experiments used the visual information scores presented in Section 3 as the measures for Max and POMDP, and the weight coefficient for the immediate reward was τ=0.8 . Figures 5(a) and 5(b) show the camera selection results for the video sequences POM Terrace1 and HUMANEVA Walk1; the best camera was selected for each frame of the

POM Terrace1 sequence on the basis of the camera quality measures described in Section 4.2, and the best camera was selected for the other sequences at intervals of 5 frames. The camera selection results indicate that there are some frequent camera switches using LYM and DBN owing to false selection because errors were introduced in the motion detection process when there are two people in the scene. Moreover, the LYM method was especially prone to frequent switching when the person utility is approximated to zero. In contrast, our method effectively predicted future trends of the visual information scores on the basis of history, and this reduced the number of false selections, resulting in smoother visual effects.



(a) POM Terrace1 selection results

(b) HUMANEVA Walk1 selection results

Figure 2: Selection results for the different methods.

5 CONCLUSIONS

Real-time selection of the most informative video stream from a number of video streams has become one of the key issues in visual analysis and processing. The experimental results show that our proposed POMDP-based method has a higher degree of accuracy and is more stable than other methods. In addition, we have proposed a visual information score function for extracting and scoring visual features associated with global motion, object properties, and special events, and this function can accurately reflect and describe the visual information in a scene.

ACKNOWLEDGEMENTS

This work is supported by The National Natural Science Foundation of China (61272219, 61100110 and 61021062), The National High Technology Research and Development Program of China (2007AA01Z334), The Program for New Century Excellent Talents in University of China (NCET-04-04605) and The Science and technology program of Jiangsu Province (BE2010072, BE2011058 and BY2012190).

REFERENCES

Soro S, Heinzelman W. 2009. A survey of visual sensor networks. Advances in Multimedia.

Li Y, Bhanu B. 2009. Utility-Based Camera Assignment in a Video Network: A Game Theoretic Framework. IEEE Sensors Journal 11(3).

Daniyal F., Taj M., Cavallaro 2010. A Content and task-based view selection from multiple video streams. Multimedia Tools and Applications ,46.

Bimbo A. D., Pernici F. 2006. Towards on-line saccade planning for high-resolution image sensing. Pattern Recognition Letters, 27(15).

Tessens L., Morbee M., Huang Lee, Philips W., Aghajan H. 2008. Principal view determination for camera selection in distributed smart camera networks. In Second ACM/IEEE International Conference on Distributed Smart Cameras.

Littman M. 2009. A tutorial on partially observable markov decision processes. Journal of Mathematical Psychology, 53(3).

Spaan M, Vlassis N. 2005. Perseus: Randomized point-based value iteration for POMDPs. Journal of Artificial Intelligence Research, vol 24.

Zhang S., Ding H., and Zhang W. 2011. Background Modeling and Object Detection Based on Two-Model. Journal of Computer Research and Development,48(11).

Hu W. M., Hu M., Zhou X., et al. 2006. Principal axis-base correspondence between multiple cameras for people tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(4).

Laptev I. 2005. On space-time interest points. International Journal of Computer Vision. Vol 64.

Fleuret F., Berclaz J., Lengane R., Fua P. 2008. Multi-camera people tracking with a probabilistic occupancy map. IEEE Transaction on Pattern Analysis and Machine Intelligence, vol 30(2).

Leonid S., Alexandru O., Michael J. 2010. HUMANEVA: Synchronized Video and Motion Capture Dataset and Baseline Algorithm for Evaluation of Articulated Human Motion. Journal of Computer Vision.