Dynamic Camera Scheduling for Visual Surveillance in Crowded Scenes using Markov Random Fields

João Neves, Hugo Proença
IT - Instituto de Telecomunicações
jcneves@ubi.pt, hugomcp@di.ubi.pt

Abstract

The use of pan-tilt-zoom (PTZ) cameras for capturing high-resolution data of human-beings is an emerging trend in surveillance systems. However, this new paradigm entails additional challenges, such as camera scheduling, that can dramatically affect the performance of the system. In this paper, we present a camera scheduling approach capable of determining - in real-time - the sequence of acquisitions that maximizes the number of different targets obtained, while minimizing the cumulative transition time. Our approach models the problem as an undirected graphical model (Markov random field, MRF), which energy minimization can approximate the shortest tour to visit the maximum number of targets. A comparative analysis with the state-of-the-art camera scheduling methods evidences that our approach is able to improve the observation rate while maintaining a competitive tour time.

1. Introduction

The co-existence of humans and video surveillance cameras in outdoor environments is becoming commonplace in modern societies. This new paradigm has raised the interest in automated surveillance systems capable of acquiring biometric data for human identification purposes. Considering that these systems are aimed at covering large areas, the use of PTZ-based systems is a popular choice, since the mechanical properties of these devices allow to zoom-in on arbitrary scene locations. In such systems, a master-slave configuration is usually adopted [12]. The master camera is responsible both for detecting and tracking subjects in the scene, so that it can instruct the active camera to point to specific locations. In these scenarios is quite common that the number of targets exceeds the available active cameras, which demands the use of a schedule technique to maximize the number of targets imaged and the number of shots taken from each one.

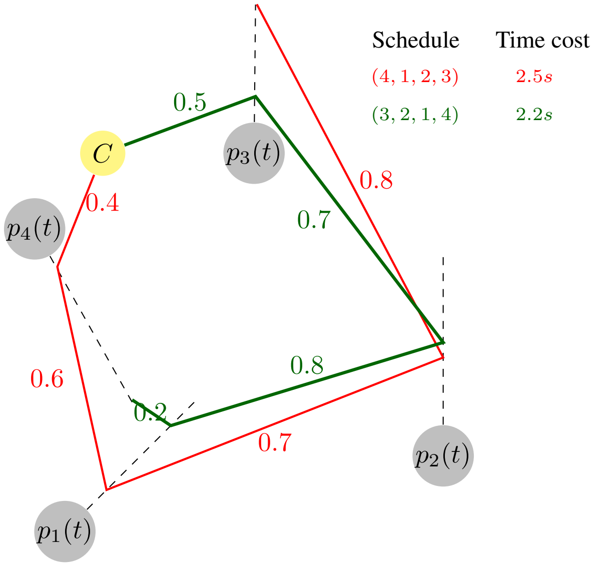


Figure 1. Illustrative example of the camera scheduling problem: given the actual and the estimated locations of a set of four targets in the scene, the goal is to determine the tour that a PTZ camera (C) should perform to observe every target while minimizing the cumulative transition time.

In this paper, we argue that a good schedule technique should plan the order to visit each target in the minimum amount of time, in order to start the acquisition process as soon as possible and maximize the number of samples taken from the subjects in the scene. As noted in previous works, the exhaustive solution for this problem is $O(N!)$, N being the number of targets in the scene. Although this brute-force strategy is feasible for a reduced number of targets, the real-time nature of this problem prohibits the use of an exhaustive search for more than six targets [2]. Accordingly, we propose a MRF-based approach to estimate an approximate solution in real-time. When compared to previous works, this formulation has two major advantages:

1. it is able to determine an approximate solution in less

than 30 ms for 15 targets;

2. is general enough to accommodate multiple cost functions. In this work, our MRF model incorporates both the transition cost and targets deadlines, in order to determine the shortest tour that observes the maximum number of targets. However, the proposed model can be easily customized to specific scenarios (e.g., prioritize frontal faces).

To demonstrate the validity of our approach, we have carried out a performance comparison with the state-of-the-art camera scheduling approaches using targets walks generated from real-world data to ensure realistic and plausible human paths. The observed results show that the proposed approach is able to successfully observe more targets in time similar to the most competitive state-of-the-art methods.

Contributions: 1) The dynamic scheduling of PTZ cameras is modelled as a graph-based problem whose solution can be approximated by the energy minimization of a MRF; 2) A realistic virtual simulation for camera scheduling evaluation purposes is described, i.e. we build on the works [1] and [15] to generate realistic and plausible human trajectories; 3) A comparative analysis with state-of-the-art camera scheduling algorithms evidences that our approach improves the observation rate while maintaining a competitive tour time.

Organization: A review of the existing camera scheduling approaches is outlined in section 2. Section 3 presents the proposed approach and section 4 describes the virtual path generation technique. The experimental results are presented in section 5 and conclusions are outlined in the section 6.

2. Related Work

Camera scheduling in PTZ-based systems can be broadly divided in coverage and saccade approaches. In the former, the cameras are set in an intermediate zoom state so that multiple targets are observable by the same device. The goal is to maximise the number of targets seen by the complete set of cameras [16, 9, 7].

On the contrary, in a saccade approach each camera just observes one target at a time. A sequence of saccades is planned, in real-time, to maximize the number of different targets observed and minimize the cumulative transition time. Some works have presented solutions to variants of this problem [10], but Costello *et* al. [4] were the first to explicitly define and propose a solution to this problem. Considering the similarities with network packet routing problem, the authors proposed the use of the Current Minloss Throughput Optimal (CMTO) to schedule a set of observations. Targets weights were determined by their residual time to exit the scene and the observation sequence was constructed by minimizing the expected weighted loss, i.e the sum of targets weights not observed. Bimbo and Pernici [2] addressed the problem by modelling it as the kinetic travelling salesman problem (KTSP), an extension of the classical travelling salesman problem where the cities positions change over time. However, this problem has not a known solution that runs in polynomial time, which restrains its use in real-time scenarios. To address this issue the KTSP is solved, by exhaustive search, for the six targets with the shortest deadlines. A similar strategy was used in [13], where a greedy best-first search was employed to determine the optimal plan. Qureshi and Terzopoulos [14] relied on greedy algorithms such as the Shortest Elapsed Time First and weighted Round Robin (RR). The weighted RR is able to efficiently distribute targets to different cameras, however, at each camera, the waiting list was scheduled based on a multi-class first-come first-served (FCFS) policy, i.e. the class was determined by the number of times the person had been imaged. In [8] the best-first heuristic was advocated as a good approximation to dynamically estimate new observation plans. Targets were modelled as graph nodes and transition costs were defined according to their distance to the camera and expected time to exit the scene. Lim et al. [11] constructed a directed acyclic graph based on the starting time of 'task visibility intervals', which were inferred by prediction. The scheduling problem was formulated as a maximal flow problem and a dynamic programming scheme was proposed to solve it. Ilie and Welch [6] relied on a greedy algorithm to determine a plan based on geometric reasoning.

3. Proposed Method

As Figure 2 illustrates, the proposed model is composed by N vertices, which represent the position of each target in the sequence of saccades. Also, each vertex can be assigned to N different labels, corresponding to the N targets in the scene. This structure allows to determine the order that each target will be observed by taking into account both the temporal constraints (vertex information) and the transition costs (pairwise relations between vertices).

Let $G = (V, E)$ be a graph representing a MRF, composed of a set of t_v vertices V, linked by t_e edges E. The MRF is a representation of a discrete latent random variable $L = \{L_i\}, \forall i \in V$, where each element L_i takes one value l_u from a set of labels.

In this problem, a MRF configuration $l = \{l_1, ..., l_{t_v}\}$, determines an acquisition sequence of N targets. Besides, we define G to be a complete graph, whose edges encode the cost of assigning the target l_u to the i^{th} position and the the target l_v to the j^{th} position. The edges between con-

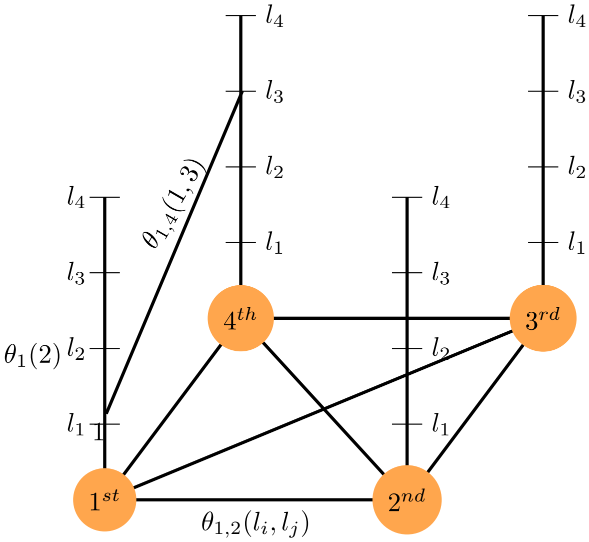


Figure 2. Illustrative example of the MRF used in our approach when four targets are in the scene. Labels encode the set of targets in the scene, whereas the nodes correspond to the order that they will be imaged.

secutive vertices correspond to the transition cost of moving the camera from the target u to v, whereas the edges of non-consecutive vertices are used to avoid repetitions in the sequence of observations. The energy of a configuration l of the MRF is the sum of the unary $\theta_i(l_u)$ and pairwise $\theta_{i,j}(l_u, l_v)$ potentials:

$$E(l) = \sum_{i \in \mathcal{V}} \theta_i(l_u) + \sum_{(i,j) \in \mathcal{E}} \theta_{i,j}(l_u, l_v). \tag{1}$$

According to this formulation, determining the best tour is equivalent to infer the random variables in the MRF by minimizing its energy:

$$\hat{l} = \arg\min_{l} E(l), \tag{2}$$

where $\hat{l_1}, ..., \hat{l_{tp}}$ are the targets index. As an example, if four targets are considered, the configuration $\{2,3,1,4\}$ determines p_2 as the first subject to be visited, p_3 as the second, and so on.

In this paper, the MRF was optimized according to the Loopy Belief Propagation [5] algorithm. Even though it is not guaranteed to converge to global minimums on loopy non-submodular graphs (such as our MRF), we concluded that the algorithm provides good approximations (refer to Section 5).

3.1. Unary and Pairwise Potentials

We first define the notation used to describe the proposed approach.

- $p_u(t) = (x_u(t), y_u(t), z_u(t))$: the 3D position of the u^{th} target at time t;
- $\alpha(p_u)$: the pan angle corresponding to the cartesian coordinates of p_u;
- $\beta(p_u)$: the tilt angle corresponding to the cartesian coordinates of p_u;
- $\Lambda_u(t)$: expected time to target p_u leave the scene;
- τ : average time required to acquire a target.

In this problem the unary costs of the first vertex have been modelled as the transition cost to move the camera from the actual position to each target. Besides, targets deadline (Λ_i) is also taken into account by greatly penalizing sequences with $\Lambda_i(t) < \epsilon$ in last vertices:

$$\theta_i(l_u) = \begin{cases} \mathcal{K}\Big(\alpha(C) - \alpha(p_u(t)), \beta(C) - \beta(p_u(t))\Big), & \text{if } \Lambda_i(t) > \epsilon, \\ 0, & \Lambda_i(t) < \epsilon \text{ and } i = 1, \\ \infty, & \text{otherwise,} \end{cases} \tag{3}$$

where C is the ground cartesian coordinate to which the camera is pointing, whereas $\mathcal{K} : (\alpha, \beta) \rightarrow \Delta$ is a camera dependent function that determines the consumed time Δ to change pan and tilt values by α and β, respectively. The pairwise potential between two adjacent vertices $\theta_{i,j}(l_u, l_v)$ is defined as the time required to point the camera to p_v assuming that is pointing to p_u:

$$\theta_{i,j}(l_u, l_v) = \begin{cases} \mathcal{K}(a, b), & \text{if } u \neq v \text{ and } c(u,v) = 1, \\ 0, & \text{if } u \neq v \text{ and } c(u,v) = 0, \\ \infty, & \text{otherwise,} \end{cases} \tag{4}$$

where $a = \alpha(p_u(t + \tau * i)) - \alpha(p_v(t + \tau * j))$ and $b = \beta(p_u(t+\tau*i)) - \beta(p_v(t+\tau*j))$. The logical function c determines if u and v are two consecutive vertices. Besides, the estimation of $p_u(t+\tau*i)$ is attained by predicting targets position using a constant velocity model.

4. Virtual Path Generation

The assessment of camera scheduling performance can be carried out using two distinct strategies: 1) integration in a running automated surveillance system; 2) performing an independent evaluation using pre-acquired human walks from the tracking module of a calibrated camera. In the former case, the results may be misleading, since it is difficult to separate the performance of the control module from the overall system. On the other hand, relying on pre-acquired human walks greatly limits the number of available paths. The use of randomly generated walks can overcome dataset size limitations, but it is highly prone to generate non-plausible paths.

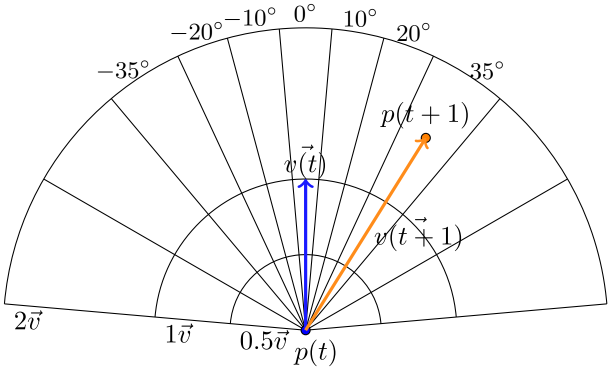


Figure 3. Illustration of the discrete grid used to model human transitions with respect to angular direction and velocity module. Adapted from [1].

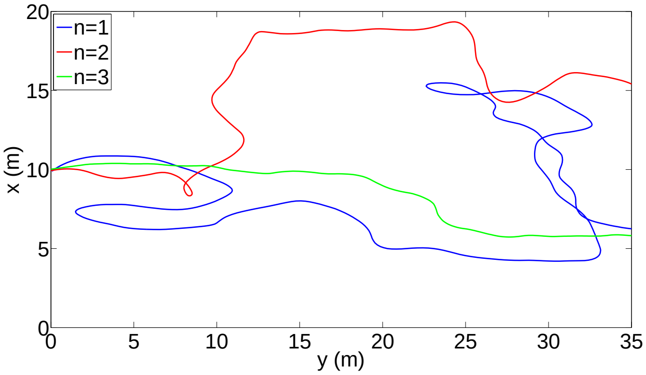


Figure 4. Examples of three virtual paths generated using the conditional distribution $p(\{\theta_t, r_t\}|(\{\theta_{t-1}, r_{t-1}\}, ..., \{\theta_{t-n}, r_{t-n}\})$ for different values of n. Note the increasingly linear shape of paths with respect to n.

Consequently, we use a virtual human walk generator to perform an independent evaluation of camera scheduling algorithms. Rather than assume constant direction and velocity model [2] - which restricts data variability - we build on the works [1, 15] to generate a virtually unlimited number of synthetic human walks.

Let $(x(t), y(t), z(t))$ be the position of a target p at the time t and $v(t)$ the velocity vector, targets movement is discretized into a grid with eleven possibilities in angular direction and three possibilities in acceleration ratio, as illustrated in Figure 3. Path generation is performed by iteratively sampling the $P(\theta, r)$ distribution to determine $p(t+1)$. In order to capture the typical behaviour of humans in surveillance scenarios, the distribution P is inferred from a set pre-acquired human walks. Additionally, we adopt the 'toward destination' behaviour - described in [15] - by dynamically re-weighting P with respect to the desired destination.

However, this strategy is memoryless, i.e., it does take in account the previous (θ, r) transitions to decide the next state, which, again, may yield non-plausible paths. To address this issue we rely on the conditional distribution $p(\{\theta_t, r_t\}|(\{\theta_{t-1}, r_{t-1}\}, ..., \{\theta_{t-n}, r_{t-n}\})$.

In our experiments, we have acquired ten paths from ten persons walking through a parking lot of 20 m by 40 m at ten frames per second (FPS) - corresponding to more than 30,000 human path positions - to infer $P(\theta, r)$.

Figure 4 illustrates the effect of n on path irregularities, such as small loops. Despite higher values could improve path reality, it would also require an higher number of training data to accurately infer the distribution p. As such, we use $n = 3$ in the evaluation of the camera scheduling algorithms.

5. Experimental Results

In this section, we evaluate the proposed approach using a virtual simulation. To replicate the conditions of a common surveillance scenario, the scene size used was similar to a typical parking lot (20 x 40 m) and the camera was assumed to be located at (0,0,5m). Also, targets paths were generated using the method described in section 4 and their initial positions were randomly selected. A Hikvision DS-2DF PTZ camera was used to estimate the function $\mathcal{K}$ in a similar fashion as in [2]. All experiments were performed in a Intel Core i7-2700K @ 3.50GHz.

```
Data: $P$, camera schedule algorithm $F$, service time $s$
Result: $t_1$,$c_1$
t=0;
$t_1$=0,$c_1$=0;
waitList=$\{1, 2, ..., \#P\}$;
while !isempty(waitList) do
    select next target $p = F(P)$;
    compute transition cost $\Delta$;
    remove $p$ from waitList;
    if $\Lambda_a(t) < \epsilon$ then
        $t_1 : t_1 + \Delta + s$;
        $c_1 : c_1 + 1$;
    end
end
```

Algorithm 1: Pseudocode for the simulation used to evaluate the performance of camera scheduling approaches.

Considering that we were interested in evaluating the time required to observe all the targets and the number of targets successfully acquired, the simulation S was defined as $S\colon P \rightarrow \{\Gamma, \Theta\}$, where $P = \{p_1(t), p_2(t), ..., p_n(t)\}$ defines the targets positions with respect to time, $\Gamma = \{t_1, t_2, ..., t_k\}$ defines the consumed time during the k^{th} acquisition tour when all the targets were in the scene, and $\Theta = \{c_1, c_2, ..., c_k\}$ the number of targets successfully acquired in each acquisition tour. To avoid disparate values of k in the same simulation for different algorithms, we opted to restrain the simulation to a single tour, i.e. $k = 1$. Algorithm 1 presents the pseudocode of the proposed simulation.

Our approach was compared to typical schedule routines adopted in [4, 14, 3], namely the FCFS and the Earliest Deadline First (EDF). Moreover, a comparison with the works [2] and [8], hereinafter designated as TDO and Krahn *et* al., was also performed.

5.1. Cost Time

The analysis of Γ with respect to N furnishes insight about the algorithms efficiency to acquire a set of N targets. Figure 5 depicts the results attained using 100 simulations for up to 15 targets. Regarding the comparison with naive schedule approaches - Figure 5a) - it is evident that the MRF-based algorithm can acquire a set of N persons faster, allowing the camera to repeat the acquisition sooner and thus collect more pictures. When considering the comparison with the work of Bimbo and Pernici [2] - Figure 5b) - it is worth noting that our approach is unable to improve TDO results up to $N = 6$. This is explained by the six element queue used to prioritize targets with the shortest deadlines and the use of an exhaustive search to determine the best solution for this subset. As compared to the algorithm of Krahnstoever *et* al. [8], the improvements can be explained by the assumption that the best target is the one with the lowest transition cost. Even though this solution can provide good approximations, it can be improved by taking into account the positions of the remaining targets as performed in our MRF-model.

5.2. Observation Rate

Additionally, we have also evaluated the average observation rate ($\frac{\Theta}{N}$) with respect to the number N of targets in the scene. The results presented in Figure 6 clearly evidence an improvement in the number of successfully observed targets as compared to the most competitive alternatives regarding the Γ performance. This difference can be explained by the fact that the remaining approaches are mainly concerned with the minimization of tour cost.

5.3. Run-time Analysis

Considering the real-time requirement of the camera scheduling problem, we have estimated the average speed of the proposed algorithm with respect to the number of targets in the scene. For this purpose, 100 simulations were used to estimate the average running time for up to twenty targets, as illustrated in Figure 7. Our approach is capable of planning a sequence of sacades in less than 30 ms for up to 15 targets, which is residual when compared to the average time (600 ms) that the PTZ camera takes to move and acquire a shot of a target. Moreover, it is worth noting that for $N = 15$, the proposed algorithm is 10^8 times faster than an exhaustive search.

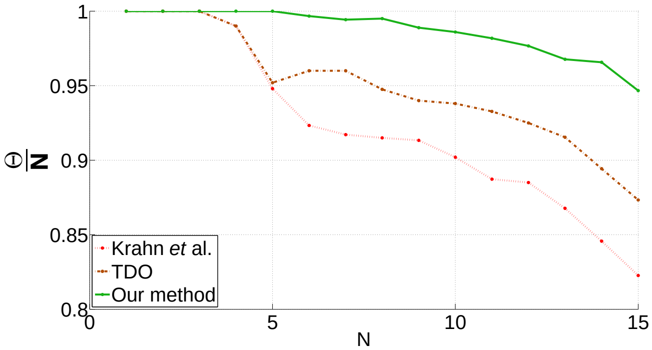


Figure 6. Comparative analysis of the average observation rate of the proposed algorithm with the most competitive alternatives regarding the Γ performance.

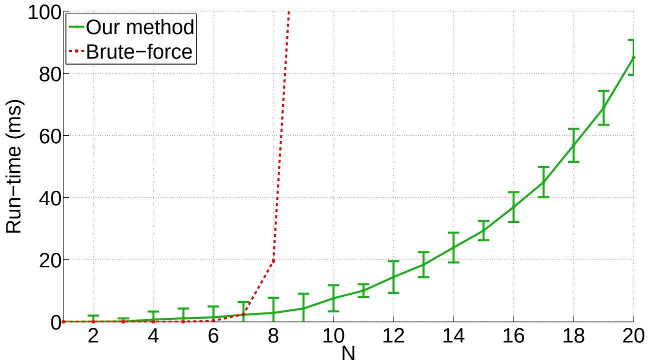
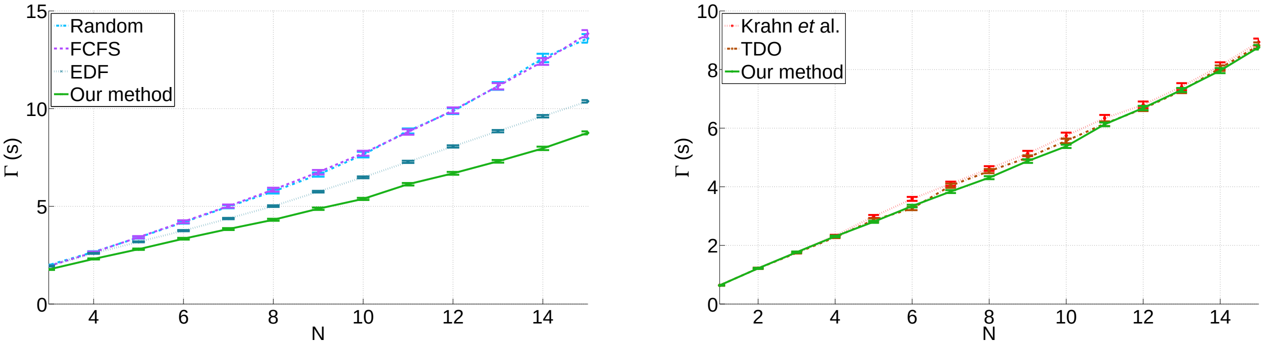


Figure 7. Average running time.

6. Conclusions

In this paper, we were concerned about the time costs of dynamic camera scheduling algorithms, which are prohibitive for crowded scenes, i.e., with over 15 subjects in the scene. Accordingly, we modelled the dynamic camera scheduling problem using a MRF model. By denoting each vertex as the position of a target in the sequence plan, our approach can take into account temporal constraints (targets deadlines) and transitions costs between consecutive vertices. The energy minimization of the MRF model yields - in real-time - a tour to acquire the maximum number of different targets while minimizing the total travel time.

Additionally, a realistic virtual simulation was proposed to assess the performance of camera scheduling algorithms. The use of realistic human walk generator - trained from real human paths - permitted to overcome dataset size constraints while maintaining the plausibility of human walks.



a) b)

Figure 5. Comparative analysis of the consumed time (Γ) required to observe N persons in the scene. a) Our approach is compared with common scheduling routines previously used in PTZ-based systems [4, 14, 3]. b) The comparison with the most competitive state-of-the-art methods is presented separately for visualization purposes.

A comparative performance analysis with state-of-the-art approaches evidences that the proposed model is able to improve the observation rate while maintaining a competitive tour time. As future work, we plan to evaluate the effect of more sophisticated path prediction algorithms in the performance of our model.

6.1. Acknowledgements

This work is supported by '*FCT - Fundação para a Ciência e Tecnologia*' (Portugal) through the research grant 'SFRH/BD/92520/2013', and the funding from 'FEDER - QREN - Type 4.1 - *Formação Avançada*', co-founded by the European Social Fund and by national funds through Portuguese 'MEC - *Ministério da Educação e Ciência*'.

References

[1] G. Antonini, M. Bierlaire, and M. Weber. Discrete choice models of pedestrian walking behavior. *Transportation Research Part B: Methodological*, 40(8):667 – 687, 2006.

[2] A. D. Bimbo and F. Pernici. Towards on-line saccade planning for high-resolution image sensing. *Pattern Recognition Letters*, 27:1826 – 1834, 2006.

[3] Y. Cai, G. Medioni, and T. B. Dinh. Towards a practical ptz face detection and tracking system. In *IEEE Workshop on Applications of Computer Vision (WACV)*, pages 31–38, 2013.

[4] C. J. Costello, C. P. Diehl, A. Banerjee, and H. Fisher. Scheduling an active camera to observe people. In *Proceedings of the ACM 2nd International Workshop on Video Surveillance and Sensor Networks*, pages 39–45, 2004.

[5] P. Felzenszwalb and D. Huttenlocher. Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1):41–54, 2006.

[6] A. Ilie and G. Welch. Online control of active camera networks for computer vision tasks. *ACM Trans. Sen. Netw.*, 10(2):25:1–25:40, Jan. 2014.

[7] N. Kariotoglou, D. Raimondo, S. Summers, and J. Lygeros. A stochastic reachability framework for autonomous surveillance with pan-tilt-zoom cameras. In *50th IEEE Conference on Decision and Control and European Control Conference*, pages 1411–1416, 2011.

[8] N. Krahnstoever, T. Yu, S.-N. Lim, K. Patwardhan, and P. Tu. Collaborative Real-Time Control of Active Cameras in Large Scale Surveillance Systems. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications - M2SFA2*, 2008.

[9] Q. Li, Z. Sun, S. Chen, and Y. Liu. A method of camera selection based on partially observable markov decision process model in camera networks. In *American Control Conference*, pages 3833–3839, 2013.

[10] S.-N. Lim, L. Davis, and A. Elgammal. Scalable image-based multi-camera visual surveillance system. In *IEEE Conference on Advanced Video and Signal Based Surveillance*, pages 205–212, 2003.

[11] S.-N. Lim, L. Davis, and A. Mittal. Task scheduling in large camera networks. In *Proceedings of the of the Asian Conf. on Computer Vision*, pages 397–407. 2007.

[12] J. Neves, J. Moreno, S. Barra, and H. Proença. A calibration algorithm for multi-camera visual surveillance systems based on single-view metrology. In *Proceedings of the 7th Iberian Conference (IbPRIA)*, pages 552–559. 2015.

[13] F. Qureshi and D. Terzopoulos. Planning ahead for ptz camera assignment and handoff. In *Proceedings of the International Conference on Distributed Smart Cameras*, pages 1–8, 2009.

[14] F.-Z. Qureshi and D. Terzopoulos. Surveillance camera scheduling: a virtual vision approach. *Multimedia Systems*, 12(3):269–283, 2006.

[15] T. Robin, G. Antonini, M. Bierlaire, and J. Cruz. Specification, estimation and validation of a pedestrian walking behavior model. *Transportation Research Part B: Methodological*, 43(1):36 – 56, 2009.

[16] Y. Xu and D. Song. Systems and algorithms for autonomous and scalable crowd surveillance using robotic ptz cameras assisted by a wide-angle camera. *Autonomous Robots*, 29(1):53–66, 2010.