



HAL
open science

Two Families of Algorithms to Film Sport Events with Flying Robots

Enrico Natalizio, Rosario Surace, Valeria Loscrì, Francesca Guerriero,
Tommaso Melodia

► **To cite this version:**

Enrico Natalizio, Rosario Surace, Valeria Loscrì, Francesca Guerriero, Tommaso Melodia. Two Families of Algorithms to Film Sport Events with Flying Robots. The 10th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), Oct 2013, Hangzhou, China. pp.319-323. hal-00917521

HAL Id: hal-00917521

<https://hal.science/hal-00917521>

Submitted on 11 Dec 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Two Families of Algorithms to Film Sport Events with Flying Robots

Enrico Natalizio*, Rosario Surace†, Valeria Loscri†, Francesca Guerriero†, Tommaso Melodia‡

* Lab. Heudiasyc, UMR CNRS 7253, France. e-mail: enrico.natalizio@hds.utc.fr

† University of Calabria, Italy. e-mail: (rsurace, vlocri, guerriero)@deis.unical.it

‡ State University of New York at Buffalo, USA. e-mail: tmelodia@eng.buffalo.edu

Abstract—In this paper, we introduce two families of distributed algorithms to control the movement of groups of flying robots that are monitoring an event by moving over the field where the event takes place, while optimizing some specific objective. In order to show the effectiveness of our algorithms, we formulate the *Sport Event Filming (SEF)* problem. The objective of the problem is to maximize the satisfaction of event viewers while minimizing the distance traveled by the camera-drones. We propose two families of solutions to solve the dynamic version of the problem, where the flying robots do not have any knowledge of the input sequence and move in reaction to the movements of the protagonists of the event. The first family (Nearest Neighbor) is based on a technique used in robotic systems, whereas the second family (Ball Movement Interception) is designed based on specific characteristics of the SEF problem. We present extensive simulation results for both families in terms of average viewer satisfaction and traveled distance for the flying robots, when several parameters vary.

Index Terms—Sport Event Filming (SEF) problem, Flying robots, VRP with Soft Time Windows

I. INTRODUCTION

Flying robots, also known as Unmanned Aerial Vehicles (UAV) or drones, are aerial vehicles that operate without a human pilot. Flying robots are usually equipped with a positioning system, storage memory, and a wireless transceiver. They can fly at considerable speed, 60 km/h for commercial devices and 220 km/h for military aircrafts. Since their creation, flying robots have found many uses in civil and military applications. Currently, they are most often used for aerial reconnaissance, scientific research, logistics and transportation, or more in general, in all the situations where a direct human intervention would be hazardous. A brilliant example of flying robots' usefulness was presented in Fukushima in 2011, when a flying robot was used to explore the disaster site at Japan's devastated nuclear power plant. We are convinced that the real potential of flying robots consists in achieving coordination and cooperation among the devices of a fleet, and that the correct design of coordination/cooperation schemes would pave the way for the realization of mission-oriented devices.

In this paper, we make a step in the direction of deploying coordination/cooperation schemes for flying robots operations

*This work has been carried out in the framework of the Labex MS2T, which is funded by the French Government, through the program "Investments for the future", managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02).

by proposing, formulating and simulating the *Sport Event Filming (SEF)* problem. We introduce this problem in order to provide a novel application scenario, where we can develop strategies to coordinate the movement of a group of mobile robots in the presence of highly varying time-space constraints to film/monitor a sequence of actions while optimizing some specific objective. Nevertheless, a solution to this problem is of interest for several application domains. Besides TV filming, it would be beneficial for environmental monitoring, disaster recovery, site inspection and exploration, etc.

Specifically, the SEF problem copes with the organization of a fleet of flying robots able to fly over a limited field to film a sport event with the objective of maximizing the satisfaction experienced by viewers who watch the game on TV, while minimizing the traveled path.

The family of problems we deal with are usually referred to as Dynamic Vehicle Routing (DVR) problem, and the static variants taken into consideration in this work are all NP-hard problems. Specifically for the event filming problem some solutions have been proposed [2], [7], [8], [4]. The main disadvantage of these solutions is that cameras are fixed. Therefore, they are not applicable to the more general problem of coordinating flying robots movements to film an event in a hostile or hazardous environment. Furthermore, they cannot provide the same level of accuracy or entertainment given by mobile devices. Whereas several solutions exist for mobile sensor networks in static scenarios, to the best of our knowledge, no schemes using flying robots have been proposed to solve this specific and dynamic problem.

The core contributions of our work can be outlined as follows:

- we describe and formulate an interesting and unexplored problem in the framework of self-organization of mobile video sensing devices;
- while in a previous work we proposed a mathematical model for the *static* version of the problem [3], in this work, we propose two families of algorithms to solve the *dynamic* version of the problem in a distributed way and without any knowledge of the sequence of actions.

The rest of this paper is organized as follows. Section II presents the Vehicle Routing Problem and its variants. In Section III we propose two families of distributed techniques for the optimal placement of flying robots. These schemes are tested and analyzed through several simulation campaigns in Section IV. Finally, Section V concludes this paper.

II. RELATED WORKS

The problem of determining the movement pattern for a certain number of flying robots when they have to film an event while maximizing viewer satisfaction and minimizing the total transportation costs can be considered as a special case of the *Vehicle Routing Problem with Time Windows* (VRPTW). Specifically, it can be classified as a *Vehicle Routing Problem with Soft Time Windows* (VRP-STW), where the sequence of points in the field to be filmed represent the customer to be served. If a specific point is not timely filmed, this affects only the satisfaction of the viewers without invalidating the overall solution.

The VRPTW assume that the cities to be visited (actions in the SEF problem) are known *a priori* and will not change during the execution of the solution. However, in real applications this assumption may be too strict. In reality, we have locations to be served that can be highly variable [5]: they can be born and die at any moment, their demands can change over time even when the solution has already been calculated. Also in the SEF problem, the position to film and the time to film that location change action by action.

The problem of planning routes through service demands that arrive during a mission execution is known as the *Dynamic Vehicle Routing Problem* (DVRP) [6], because part or all the locations to reach are not known *a priori*.

In [6], the authors identify three main approaches to address DVR problems. The first approach is to simply re-optimize every time a new event takes place; in the second approach, routing policies are designed to minimize the worst-case ratio between their performance and the performance of an optimal offline algorithm that has *a priori* knowledge of the entire input sequence; in the third approach, the routing problem is embedded within the framework of queueing theory and routing policies are designed to minimize typical queueing-theoretical cost functions such as the expected waiting time in the system for the demands.

Both families of distributed algorithms we present in this work follow the first approach. In the second family, we additionally consider specific characteristics of the problem to forecast the next locations to be covered.

III. DISTRIBUTED ALGORITHMS FOR DYNAMIC VRP-STW

If the whole event sequence is available *a priori*, then the SEF problem becomes a VRP-STW problem, which has been modeled and solved to optimality [3]. Since this assumption is not realistic for a real-time system, new optimization methods need to be designed to tackle the dynamic version of the SEF problem.

In Section II we mentioned that three different approaches have been identified. The first of these approaches simply proposes to re-optimize every time a new event takes place. This approach is the most suited for the specific communication and movement capabilities of the flying robots to offer a feasible and practical solution to the event filming problem. In fact, the sub-optimal solution will be computed action-by-action by the flying robots that cooperate by exploiting their communication capabilities in a distributed and self-organized fashion. For

this purpose, we introduce in the distributed strategies the *coordination time*, T_{coord} , which is the time needed by the robots to communicate with each other and determine which of them will move to follow the newly generated action.

In the following we present two families: Nearest Neighbor (NN) and Ball Movement Interception (BMI), each of them consisting of four different distributed techniques to solve the event filming problem.

A. Nearest Neighbor

The *Nearest Neighbor* technique for DVR problems in robotic system is presented in [1]. The core idea is that viewer satisfaction increases when a flying robot is able to reach the location of the current action as quickly as possible, and that the minimum traveled distance is achieved by the closest flying robot. Thus, the flying robot that is the closest to the location of the action is the one chosen to move and film the action. The following three techniques are extensions of the basic NN technique.

B. Nearest Neighbor-Division Field

A disadvantage of the NN technique is that when a sequence of actions occurs in a limited area, the same flying robots will be chosen to film it. If the duration of this sequence extends over time, it would cause one robot to reach its maximum feasible traveled distance much earlier than the others.

Based on these considerations, we introduce the *Nearest Neighbor-Division Field* (NN-DF). In the NN-DF technique, each robot is assigned to a portion of the field, and it will film the actions that are located inside that portion.

This technique has the disadvantage of not choosing the robot that is the nearest to the current action, which can result in a reduced satisfaction for the viewer. We will see in Section IV the effects of this with respect to the reduced area for each robot to monitor.

C. Nearest Neighbor with Specular Repositioning

In the previous two techniques only one robot moves when a new action is born. The *Nearest Neighbor with Specular Repositioning* (NN-SR) technique considers robots as belonging to a pair. When one of them, k , is chosen to move to film an action for which it is the nearest neighbor, the robot that is closest to the position specular to the action position, \bar{k} , moves as well to mirror the movement of the first. More precisely, let L and W be the length and the width of the field. When robot k moves to the position of the new action (x_a, y_a) , \bar{k} will move to $(L - x_a, W - y_a)$. It is worth noting that robots are not coupled at the beginning of the event, instead \bar{k} is chosen action-by-action depending on the proximity to the action specular position. We expect that this technique, which results in robots traveling more than the previous techniques, will be more reactive and timely in filming the actions so as to offer a higher satisfaction to the viewer.

D. Nearest Neighbor with Quasi-Specular Repositioning

A generalization of the NN-SR technique is the *Nearest Neighbor with Quasi-Specular Repositioning* (NN-QSR). The NN-SR technique makes pairs of robots move specularly. As we have already highlighted, this behavior can lead to a quick depletion of the maximum allowed traveled distance, due to the specular movements of the robot (\bar{k}) that is not filming any action. Thus, the idea behind the NN-QSR is to make the center of the field be an attractor for \bar{k} while it is repositioning in the direction of k 's specular position.

The attraction strength on the movement can be modulated through an appropriate *detour factor*, $0 \leq \beta \leq 1$. When $\beta = 0$, no detour is applied on the movement of \bar{k} , which moves to the specular position in respect of the current action position, and NN-QSR coincides with NN-SR. When $\beta = 1$, \bar{k} is completely detoured towards the center of the field. For intermediate values between 0 and 1, \bar{k} move on a point on the straight line between these two extreme points. More precisely, if (x_a, y_a) , L , W are the positions of the new action, the length and the width of the field, respectively, then \bar{k} will move to $(L \cdot (1 - \frac{\beta}{2}) - x_a \cdot (1 - \beta), W \cdot (1 - \frac{\beta}{2}) - y_a \cdot (1 - \beta))$.

By detouring the movement of \bar{k} , we expect a higher satisfaction of the viewer as compared to the NN and NN-DF techniques, without introducing a high traveled distance expenditure as in the NN-SR technique.

E. Ball Movement Interception

All the previous techniques work well if t_{birth} and t_{start} are sufficiently far in time to allow a robot to reach the action location before t_{start} . In fact, these techniques try to solve the dynamic version of the proposed problem simply by adapting as quick as possible the position of one (or one pair of) robots. None of them try to forecast the location to film for next action before its t_{birth} . As we described in [3], the static model introduces the time of "flight" of the ball when the ball is not possessed by any player, T_{fly} . This interval of time between t_{stop_i} and $t_{birth_{i+1}}$ could be used to forecast the location of next action.

We can realistically assume that robots, which are able to constantly detect the ball and its location, are also able to easily compute their trajectory. For the sake of simplicity, in this work, we consider only that the ball moves along straight lines. We consider the parabolic trajectory of the ball as flatted on the straight line lying on the game field plane, and we do not take into consideration special effects that can be given to the ball.

By assuming that robots know the trajectory of the ball, they can estimate the next player who will hold the ball. Through this estimation, before the ball reaches the next player they can start moving towards the straight line between the position of the previous action and that of the player expected to receive the ball. Thus, we introduce a new family of techniques, called *Ball Movement Interception* (BMI), which includes all the previous techniques augmented by this knowledge: *Ball Movement Interception* (BMI), *Ball Movement Interception with Division Field* (BMI-DF), *Ball Movement Interception with Specular Repositioning* (BMI-SR) and *Ball Movement*

Interception with Quasi-Specular Repositioning (BMI-QSR). It is important to note that we do not assume that unexpected interceptions of the ball destined to a specific player are neglected. In fact, such events would simply cause a degradation in the performance of this family of techniques.

IV. PERFORMANCE RESULTS

In this Section we will show two simulation campaigns illustrating selected results obtained for the proposed algorithms when several parameters vary. We consider the *average viewer satisfaction* as the output parameter for assessing the quality of the route chosen for the robots, and the *total traveled distance* as the output parameter representing the cost of the route. In the first simulation campaign we study the impact of the detour factor, β , on the performance of the Specular Repositioning techniques. The second simulation campaign is a more general comparison among the different distributed techniques. The results have been achieved by using MATLAB 7.9.0.529 (R2009b), and they have been averaged over 1000 runs with a confidence interval of 95%. The parameters presented in Table I are used in all the simulation campaigns, specific differences will be highlighted in each campaign subsection.

We simulate the behavior of the algorithms when the number of actions in the event and the duration of each action vary, respectively. The number of actions is useful to characterize the time-space variability of the actions in the event, whereas the duration of an action represents the dynamicity of the event. Both these input parameters depend on the kind of sport that has to be filmed and their characterization is left as a future work. In our simulations, we also used a variable number of robots ($2 \div 6$), but, for matter of space, we will be able to show few results of scenarios with 2 and 4 robots.

Parameter	Value
Size of the game field ($L \times W$)	$110 \times 80 [m^2]$
Max Distance Feasible by Robots	$65 [km]$
Speed of Robots	$15 [m/s]$
Action Min Duration ($t_{birth} \rightarrow t_{stop}$)	$0.2 [s]$
Ball Min and Max Speed	$\{1 \div 40\} [m/s]$
Coordination Time (T_{coord})	$0.2 [s]$
Max Satisfaction (S_{max})	1
Actions Spatial and Temporal Distribution	random
Number of run for each scenario	1000

TABLE I: Fixed parameters used for all simulations

A. Performance evaluation varying detour factor

In this simulation campaign we want to investigate the impact of the detour factor on the performance of the QSR techniques. Hence, we compare the results of NN-QSR and BMI-QSR, when the detour factor, β , varies in the range $\{0 \div 1\}$. We let the number of actions in the event and the duration of an action vary, as shown in Table II. The range considered for the former parameter has been increased to match the time duration of a real event. The performance of the two techniques in terms of average viewer satisfaction for different number of actions is reported in Fig. 1, and traveled distance for different maximum durations of the actions in Fig. 2.

Number of Robots	2
Robot k Position	$\{(-1)^k \frac{L}{4} + \frac{L}{2}, \frac{W}{2}\}$
Action Max Duration ($t_{birth} \rightarrow t_{stop}$)	$\{2, 6, 10\}$ [s]
Detour Factor (β)	$\{0 \div 1\}$
Number of Actions	$\{100, 500, 1000\}$

TABLE II: Simulation parameters used for simulating NN-QSR and BMI-QSR

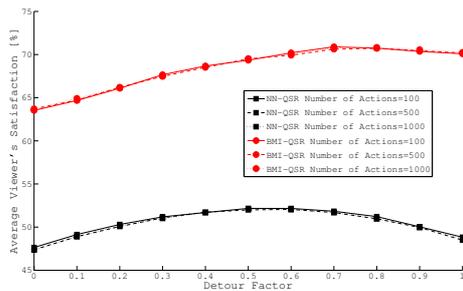


Fig. 1: QSR techniques: average viewer's satisfaction when the detour factor and the number of actions vary

From Fig. 1 we observe that use of the ball movement interception does limit the need to reposition the robot specular to the robot that is filming the action. The BMI technique leads to a high viewer's satisfaction when the attraction strength towards the center of the field increases, whereas the NN-QSR technique presents a maximum when the detour factor is between 0.5 and 0.6. This also means that different detour strengths should be applied depending on the used technique. As expected, the viewer satisfaction experienced with the BMI technique is higher on average.

The same behavior for the NN-QSR technique is presented in Fig. 2, where we can appreciate the existence of a minimum in the distance traveled by the robots when the detour factor is around 0.6. The BMI-QSR improves its performance when the detour factor grows until values very close to 1. It is interesting to remark that for both the techniques, a decrease in the distance traveled by the robots corresponds to an increase

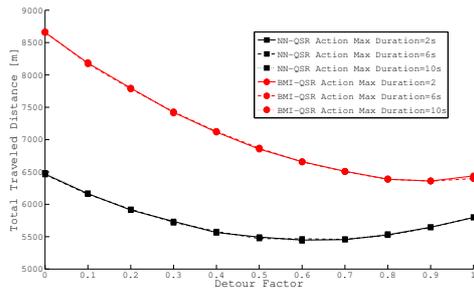


Fig. 2: QSR techniques: traveled distance when the detour factor and the action maximum duration vary

Number of Drones	4
Drone k Position	$\{(-1)^k \frac{L}{4} + \frac{L}{2}, \frac{W}{2}\}$ $\{(-1)^{\lceil k/2 \rceil} \frac{L}{4} + \frac{W}{2}\}$
Action Max Duration ($t_{birth} \rightarrow t_{stop}$)	$2 \div 10$ [s]
NN-QSR Detour Factor (β)	0.6
BMI-QSR Detour Factor (β)	0.8
Number of Actions	$\{1000 \div 5000\}$

TABLE III: Simulation parameters used for distributed algorithms comparison

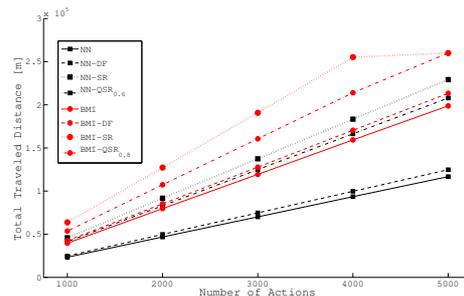


Fig. 3: Distributed algorithms comparison: total traveled distance for fixed actions maximum duration (6 [s])

in the satisfaction experienced by the viewer. Comparatively, the NN-QSR technique leads the robots to travel about 1.3 km on average less than the BMI-QSR technique, with a corresponding about 18% on average of decrease in the viewer's satisfaction. For both the output parameters, the number of actions and the action maximum duration do not significantly impact the performance of the different techniques, therefore the three simulated algorithms produce overlapping curves.

B. Comparison of Positioning Techniques

This second simulation campaign, whose main parameters are in Table III, shows the results when all the distributed techniques are applied to a scenario with a variable number of actions and a fixed action maximum duration (Fig. 3, 4) and a fixed number of actions and a variable action maximum duration (Fig. 5, 6).

In Fig. 3 we show that the distance traveled by the robots grows linearly with respect to the number of actions for all the algorithms. Thus, it is easy to predict the distance that each technique will make robots travel through an estimate of the number of actions a real event will consist of. As expected, the NN technique is the best in terms of traveled distance, both when the Division Field is used and when it is not. The basic technique of the BMI family performs as the third best for this metric, which is a very encouraging result because of the consideration we will make about the average viewer satisfaction. On average the BMI technique makes robots travel about 73 km more than NN in the considered interval. The techniques with Specular Repositioning are the worst for this metric because of the distance traveled by the robot that does not film the action.

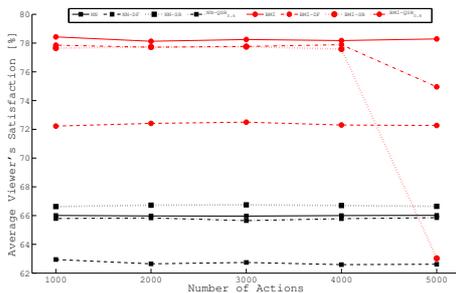


Fig. 4: Distributed algorithms comparison: average viewer's satisfaction for fixed actions maximum duration (6 [s])

The situation is reversed in Fig. 4, which shows the average viewer satisfaction. All techniques in the BMI family achieve a higher satisfaction than the corresponding techniques in the NN family. The distance between the best techniques of the two families for this parameter, which are the basic technique and the SR technique, is 14% on average. When the upper limit on the feasible traveled distance is reached, both the satisfaction achieved by BMI-SR and BMI-QSR start decreasing, since robots are not allowed to move anymore. Thus, the instantaneous satisfaction goes to zero and the average satisfaction decreases. Until the feasible distance limit is reached, the two techniques of the BMI family perform very similarly, the only main difference is that the QSR let robots travel more efficiently. Instead, we can appreciate some difference in the performance of the same techniques for the NN family, the SR technique performs 2% better on average than the QSR technique.

In Fig. 5 we can appreciate the traveled distance when the maximum duration of the actions varies. We can see that all the proposed algorithms are scalable with respect to this input parameter, and the heuristics ranking is the same of that in Fig. 3. Fig. 6 shows a logarithmic growth of the average viewer's satisfaction when the actions maximum duration increases. Very quick and short actions create troubles to all the algorithms, which do not achieve more than 30% of viewer's satisfaction, whereas they perform much better and reach 90% of satisfaction when the maximum duration is the upper value. On average, the BMI techniques have a gain of 15% over the corresponding NN techniques.

V. CONCLUSION

In the context of coordination schemes deployment for mobile robots, we have introduced the SEF problem, whose objective is to maximize the satisfaction of an event viewer while minimizing the distance traveled by the camera-drones that film the event. We considered the dynamic version of the problem where knowledge of the entire sequence of actions is not assumed to be known *a priori*. The dynamic version of the Sport Event Filming can be treated as a Dynamic Vehicle Routing problem. We solved it by re-optimizing the position of the drones every time that a new action occurs. As a future

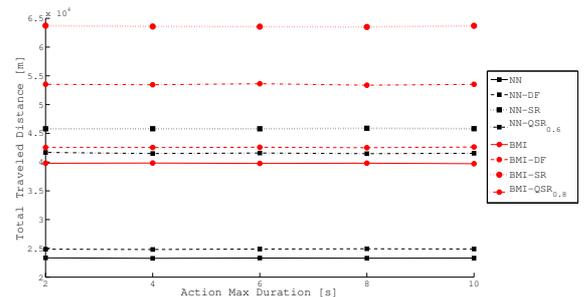


Fig. 5: Distributed algorithms comparison: total traveled distance for a fixed number of actions (1000)

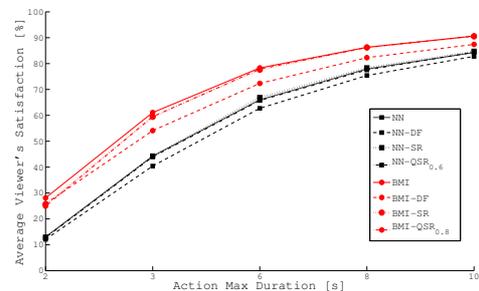


Fig. 6: Distributed algorithms: average viewer's satisfaction for a fixed number of actions (1000)

work, we will use a different approach based on queuing theoretical cost functions.

REFERENCES

- [1] F. Bullo, E. Frazzoli, M. Pavone, K. Savla, and S. L. Smith. Dynamic vehicle routing for robotic systems. *IEEE Proceeding*, 99(9):1482–1504, 2011.
- [2] F. Daniyal and A. Cavallaro. Multi-camera scheduling for video production. In *European Conference on Visual Media Production (CVMP)*, London, UK, Nov. 2011.
- [3] F. Guerriero, R. Surace, V. Loscr , and E. Natalizio. A multi-objective approach for unmanned aerial vehicle routing problem with soft time windows constraints. *Accepted for publication in Applied Mathematical Modelling*, 2013.
- [4] P. Halvorsen, S. Sægrov, A. Mortensen, D.K.C. Kristensen, A. Eichhorn, M. Stenhaus, S. Dahl, H.K. Stensland, V.R. Gaddam, C. Griwodz, and D. Johansen. In *Proc. of the International Conference on Multimedia Systems (MMSys)*, pages 48–59, Oslo, Norway, February/March 2013.
- [5] P. Kilby, P. Prosser, and P. Shaw. Dynamic vrps: A study of scenarios. *University of Strathclyde Technical Report*, (APES-06-1998):1–11, 1998.
- [6] M. Pavone. *Dynamic Vehicle Routing for Robotic Networks*. PhD thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2010.
- [7] F. Z. Qureshi and D. Terzopoulos. Surveillance in virtual reality: system design and multi-camera control. *Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR 07*, pages 1–8, June 2007.
- [8] X. Zhou, R.T. Collins, T. Kanade, and P. Metes. A master-slave system to acquire biometric imagery of humans at distance. In *ACM International Workshop on Video Surveillance*, pages 113–120. ACM Press, 2003.