Workshop on Resource Aware Sensor and surveillance NETworkS (RAWSNETS) in conjunction with 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance, 2011

Resource-Aware Sensor Selection and Task Assignment

Bernhard Dieber and Bernhard Rinner
Alpen-Adria-Universität Klagenfurt
Klagenfurt, Austria
{firstname.lastname}@aau.at

1. Introduction

Multimedia sensor networks [1] and visual sensor networks (VSN) [3] have been increasingly studied in recent years. However, the aspect of resource-awareness has just recently moved into the focus of research interests. Especially energy-aware systems that may be deployed in areas without fixed infrastructure have only recently achieved attention.

In the SRSnet (Smart Resource-aware Sensor Network) project[1], we consider smart cameras that operate autonomously within a visual sensor network. The cameras have onboard processing facilities and can execute different video surveillance tasks [4]. The sensors are connected via a low-power, wireless channel. The goal is to provide a high surveillance quality while minimizing energy consumption to prolong the network lifetime. This includes minimizing the communication effort and processing load on nodes. We also aim for selectively switching off sensors and nodes to save energy. Thus, we must perform sensor selection to find a minimal set of sensors to perform a certain task as well as task assignment to find resource-minimal assignments of tasks to nodes.

2. Problem Modeling

In our work we model the visual sensor network as a set of static cameras with associated locations and fields of view as shown in Figure 2. Additionally, we consider a set of observation points which represent areas of special interest to surveillance. These observation points pose certain surveillance requirements expressed in terms of pixels on target, frame rate and surveillance activity. We try to solve the problem of assigning each observation point to a camera of the visual sensor network in a way that the requirements of the observation points are fulfilled while the energy consumption in the network is minimized and the resources of all cameras are not exceeded. Figure 2 shows an example of such an assignment.

[1]http://SRSnet.lakeside-labs.com

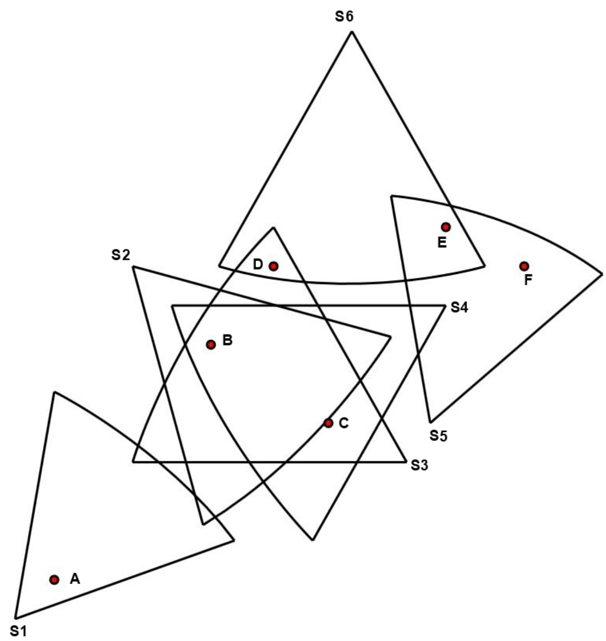

Figure 1. A visual sensor network consisting of cameras ($S1 - S6$ and observation points ($A - F$). A camera's FOV is represented as a segment.

We first developed a centralized solution that uses global knowledge on the network and the camera's resources to find a feasible assignment. This solution is based on an evolutionary algorithm and is described further in Section 3. To be able to optimize online within an already deployed network, we are developing a distributed optimization algorithm and an associated communication protocol. Our approach for this is described in Section 4.

3. Central Task Assignment

In this work we concentrate on the reconfiguration of resource-limited wireless visual sensor networks. Typically, the tasks of such a network change over time. External operators may redefine areas of interest and the surveillance

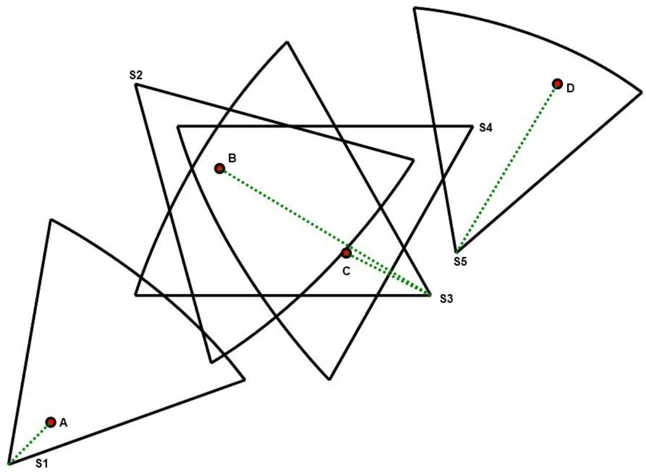

Figure 2. An example of a visual sensor network and observation points that must be covered by the cameras. Allocations of observation points to cameras are depicted by green dashed lines.

operation necessary in those areas. In SRSnet, we additionally use a subsystem for complex event detection which may trigger a reconfiguration according to certain events (e.g. tracking certain objects involved in a complex event such as "Group of persons entering a forbidden area"). The complex event detection component can refocus the attention in the network to certain objects. The audio/video subsystem is responsible for orientating the PTZ cameras and feeds orientation information into the network reconfiguration.

Resource limitations on the nodes are an important constraint in our work. The surveillance network is intended to be deployed in areas with limited infrastructure where no permanent power supply is available. To reconfigure our networks we need to *i)* select a minimal set of cameras which are able to reliably fulfill a certain surveillance task, *ii)* configure those cameras with an appropriate resolution and frame rate and *iii)* assign the necessary monitoring tasks to the nodes. Thus, we try to solve a *combined sensor selection and resource allocation* problem.

All three steps must be performed with respect to energy usage and delivered surveillance QoS. We express QoS as frame rate, pixels on target and surveillance activity for a certain point in space. Surveillance activities may be e.g., *change detection, object detection, tracking or similar tasks*. Finding a solution within those constraints is a hard task because the search space of possible solutions is very large, but the set of feasible solutions is rather small.

Solving a combined sensor selection and resource allocation problem is a challenging task. In contrast to classical optimization problems, which optimize for a single goal, this problem has multiple optimization criteria. First, the sensor coverage must be guaranteed and maximized, second, the resource usage on the nodes must be minimized. Multiobjective optimization problems are often tackled with evolutionary algorithms (Evolutionary Multiobjective Optimization, EMO) [2]. Here, an iterative approach is employed to find a feasible solution. The problem is described as a set of properties that can be altered (also called a chromosome). In each step (called an epoch) the algorithm randomly alters one or more of those properties. A fitness function is then used to determine the quality of the resulting solution (i.e. the fitness). For most multiobjective optimization problems, there is no single optimal solution but rather a set of feasible solutions which must be regarded as equally good. This set is called the Pareto front [5]. We have developed a general framework for evolutionary multiobjective optimization. This framework has been released as an open source project[2]. It allows to quickly encode a single- or multiobjective problem into an evolutionary algorithm. It then performs evolutionary optimization

We developed this central algorithm as a benchmark for our later distributed algorithm in order to compare the distributed algorithm's results against the central algorithm's solutions.

4. Distributed Task Assignment

In order to perform optimization online in a visual sensor network we are developing a distributed optimization algorithm. This optimization algorithm must operate in a fast and resource-saving manner too. An evolutionary approach would be infeasible since it requires a lot of computation as well as a huge amount on information from other nodes.

For the distributed algorithm we must accept the fact that an optimal solution may not be found in all cases. An especially challenging aspect in our setting are cases when the network cannot be clearly partitioned. Such scenarios are very common in visual sensor networks with shared fields of view and are hard to solve without sharing global knowledge on all cameras (which would require the exchange of huge amounts of information among nodes).

As an example, Figure 2 shows a network, where the cameras cannot be clustered according to the observation points in their field of view. The solution for camera 1 and observation point A may be computed independently but for the remaining cameras no clear separation can be found. The cluster of cameras $2, 3, 4$ and observation points B, C is dependent from the cluster of cameras $5, 6$ and observation points D, E, F. This means that a solution for cluster $2, 3, 4$ is dependent from the solution found in cluster $5, 6$ and vice versa. In practice this means, that all cameras except 1 may need to be be involved for finding a solution although for example camera 2 and 5 have no common observation points.

This increases the communication and calculation effort

[2] http://nemo.codeplex.com

for finding a solution. Our goal is to find ways to break those interdependencies to be able to compute solutions within small clusters. We aim at an algorithm and protocol with only a small set of commands and messages. For every observation point, the cameras iteratively agree on the assignment of that point to a camera. This is done by selecting the camera that can cover that point with minimal resource effort. If this is impossible due to a high load, the cameras will try to hand off the responsibility for other points in order to free resources to cover the new point. In the example above (Figure 2), camera 6 may hand off observation point E to camera 5 in order to free resources for covering D.

5. Current and Future Work

The algorithms presented above are subject to ongoing work. We have already developed a first prototype of the central algorithm and are currently developing the distributed version. This algorithm will be part of the prototype deployment of SRSnet in the National Park "Hohe Tauern" in summer 2011. From this deployment we will gather real world data on the performance of our optimization algorithm which will be used to further refine our approach.

Acknowledgments

This work is supported by Lakeside Labs GmbH, Klagenfurt, Austria and funded by the European Regional Development Fund (ERDF) and the Carinthian Economic Promotion Fund (KWF) under grant KWF 20214/18354/27107.

References

[1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury. A Survey on Wireless Multimedia Sensor Networks. *Computer Networks*, 51:921–960, 2007. 1

[2] C. A. Coello Coello. Evolutionary Multi-Objective Optimization: a Historical View of the Field. *Computational Intelligence Magazine, IEEE*, 1(1):28 – 36, feb 2006. 2

[3] Hamid Aghajan and Andrea Cavallaro, editor. *Multi-Camera Networks*. Elsevier, 2009. 1

[4] B. Rinner and W. Wolf. Introduction to Distributed Smart Cameras. *Proceedings of the IEEE*, 96(10):1565–1575, October 2008. 1

[5] E. Zitzler, M. Laumanns, and S. Bleuler. A Tutorial on Evolutionary Multiobjective Optimization. *Metaheuristics for Multiobjective Optimisation*, 535(535):21–40, 2004. 2