Dynamic Reconfiguration in Camera Networks: a short survey

Claudio Piciarelli, *Member, IEEE,* Lukas Esterle, Asif Khan, Bernhard Rinner, *Senior Member, IEEE* and Gian Luca Foresti, *Senior Member, IEEE*

***Abstract*—There is a clear trend in camera networks towards enhanced functionality and flexibility, and a fixed static deployment is typically not sufficient to fulfill these increased requirements. Dynamic network reconfiguration helps to optimize the network performance to the currently required specific tasks while considering the available resources. Although several reconfiguration methods have been recently proposed, e.g., for maximizing the global scene coverage or maximizing the image quality of specific targets, there is a lack of a general framework highlighting the key components shared by all these systems. In this paper we propose a reference framework for network reconfiguration and present a short survey of some of the most relevant state-of-the-art works in this field, showing how they can be reformulated in our framework. Finally we discuss the main open research challenges in camera network reconfiguration.**

***Index Terms*—Camera networks, sensor reconfiguration, active vision**

I. Introduction

CAMERA networks are nowadays widely used in many different application fields, such as building surveillance, traffic monitoring, crime prevention in public areas and crowd flow analysis. Despite many of these networks are still based on CCTV-based architectures, where the sensor control and data interpretation are in charge of human operators, the amount of acquired data often requires some form of automatic processing. This can be achieved either by centralized processing of the video streams or, for better scalability, by distributed processing using smart cameras [1], [2].

Although many works have been published on the use of camera networks for various computer vision applications, few of them considered how the *configuration* of the network influences the system performance, and how reconfiguration could be used to adapt and improve the network performance [3]. The configuration space of a network is here defined as a set of sensor parameters that can actively modify the quality and amount of the acquired data. In static camera networks, these parameters include frame rate, exposure time, aperture and resolution and can be altered during normal sensor operation. The use of Pan-Tilt-Zoom (PTZ) cameras extends the configuration space with orientation and field-of-view parameters for each sensor. Finally, in networks composed by mobile camera platforms, such as robots and unmanned aerial vehicles (UAVs), the location of the sensors become a configuration parameter as well.

C. Piciarelli and G.L. Foresti are with the Università degli Studi di Udine, Italy. L. Esterle, A. Khan and B. Rinner are with the Alpen-Adria Universität Klagenfurt and Lakeside Labs, Austria. Corresponding author: claudio.piciarelli@uniud.it

Reconfiguring a network may affect both the observed areas (i.e., by adapting position, orientation and zoom parameters) and the image quality (i.e., by adapting the internal camera parameters), which directly influence the global performance of the sensing system. A proper reconfiguration policy can thus optimize the output quality according to specific criteria. This optimization of course depends on the overall goal of the system. Typical examples are networks of PTZ or mobile cameras where the goal is to monitor the largest area possible (coverage maximization) or to focus on specific targets. Another common objective is to reconfigure the internal camera sensors in order to optimize the resource consumption in terms of power, processing and bandwidth usage.

The contribution of this paper is twofold. First, we describe a novel *reconfiguration framework* for camera networks. The framework generalizes the key concepts basically shared by all reconfiguration systems. Its main aim is to clearly highlight the system components and propose a reference notation to designate them. The framework models both main goals of the reconfiguration: (i) to enable the cameras to cooperate and work towards a common goal in a distributed fashion, and (ii) to minimize overall resource consumption in the network by optimizing the tasks assigned to the cameras. Our second contribution is to present a short survey on camera network reconfiguration. The aim of the survey is not to cover all the state-of-the-art works on this topic, but rather to show how the most relevant works on network reconfiguration can be expressed in terms of our framework.

The rest of the paper is structured as follows. In section II we present the novel framework for reconfiguring smart camera networks allowing the combination of various types of cameras such as fixed, PTZ, and aerial cameras. Section III reviews the current state of the art, classifying several relevant works according to our framework. In section VI we identify recent trends and discuss open research challenges.

II. Reconfiguration Framework

Figure 1 depicts the proposed framework for dynamic reconfiguration in camera networks. A network of distributed camera nodes observes a physical environment which often exhibits complex behaviors of the involved objects of interest. A key goal of the camera network is to capture relevant

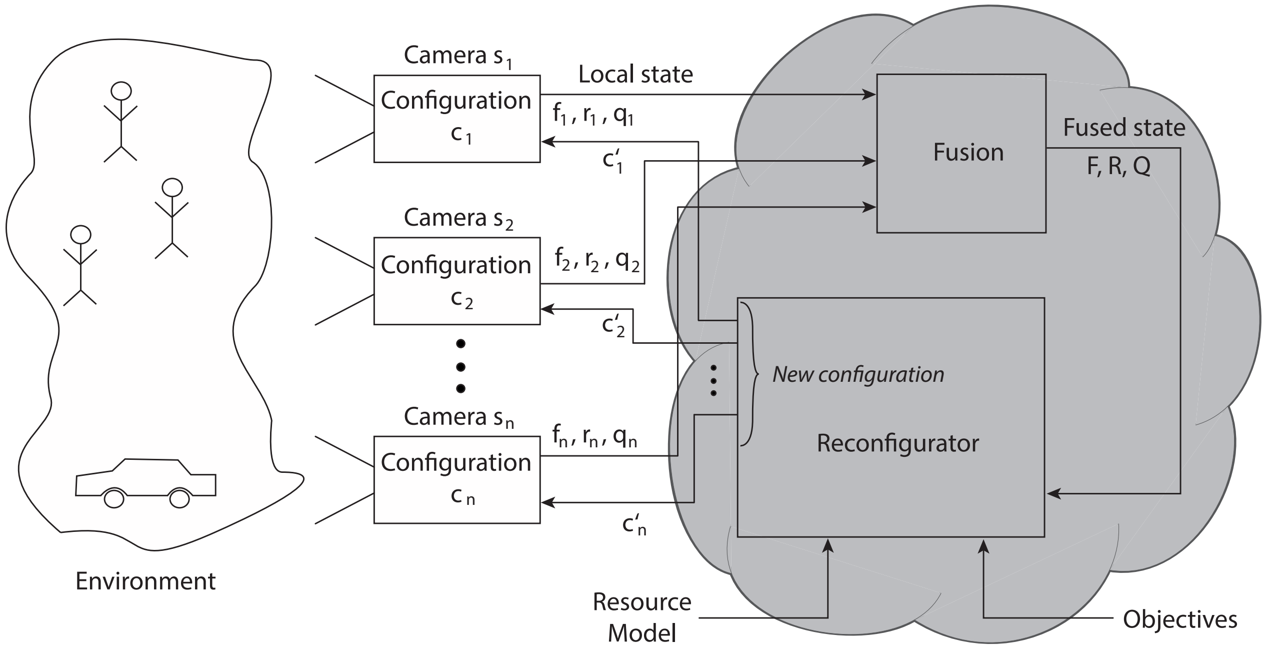

Fig. 1. High-level framework for dynamic reconfiguration in camera networks. A set of camera nodes observe the scene and derive local quality criteria and resource requirements. This local information is merged to a global state which serves as key input for the reconfigurator. It computes optimized configurations for each camera node. In any case, the reconfigurator can be either centralized or distributed among the different cameras in the network.

data of the environment and analyze it in order to provide information about the current situation. However, the level of detail and quality of the derived information depends on the functionality and available resources of the camera network. While there are methods and techniques available for reconfiguration in sensor networks, these approaches often assume omnidirectional sensors. In contrast, visual sensors imply much stricter constraints, such as directionality.

The network is composed of several camera sensors $s_i \in \{s_1, \ldots, s_N\}$. Each camera sensor performs some analysis given local sensor data and available resources of that node. The output of s_i is thus a local state f_i of the scene at a specific quality q_i with some information about the current resource usage r_i. This abstracted data may be processed in a centralized manner where each camera sensor is connected to central node. As an alternative, the camera nodes may directly exchange the data and process it in a distributed manner. In such a case, the individual cameras have to run their very own reconfigurator locally with the information received from the other cameras. An overhead from fusing all this information at the individual cameras occurs. In Figure 1 the choice between a centralized or distributed approach for information fusion and camera reconfiguration is depicted as a gray cloud. Each sensor s_i has its own configuration c_i, denoting the current state of the configurable parameters. The union of all parameters, which can be actively changed, is referred to as the configuration space of the node; the union of all node configurations is thus referred to as the configuration space of the overall network. The configuration space may include sensor resolution, frame rate, camera position, PTZ parameters and tasks to be executed on the camera.

We assume that each camera is able to compute a local quality score q_i denoting how good the current local configuration is in executing a required task at a specific state f_i. The quality thus highly depends on the overall system goal, and its measurement must be defined according to a specific application. For example, if the goal is to optimize the sensor area coverage, the size of the observed area could be a possible quality measure. Each camera also monitors its resource usage r_i, where the resource space (the set of all possible resources) is again system-dependent, and may include bandwidth usage and power consumption. Local information is then propagated in order to define a common network state F and compute a joint network quality Q with a total resource consumption R, representing an aggregated view of the current network state based on the local information from the individual cameras. The global values F, Q, and R serve as key input to the reconfigurator, which computes new configurations $c'_1 \ldots c'_N$ for the camera nodes based on some global objective on quality and/or resource consumption.

Figure 1 represents the high-level information flow for dynamic reconfiguration. As the reconfiguration should allow the network to deal with the dynamics of the environment, the information exchange and reconfiguration has to be executed in regular intervals. Our framework helps to identify the key processes of the system:

a) Local analysis: The local analysis of each sensor is dependent on the system goal. It generally processes the video stream to detect changes, track moving objects, perform high-level behavior interpretation etc.—just to name typical examples. There is a plethora of methods available for computing the local analysis from a single sensor input data. By abstracting the output of local processing with f_i, q_i and r_i, network reconfiguration becomes independent of the local analysis.

b) Information distribution and fusion: In general, the local states f_i, the quality scores q_i and resource consumptions r_i of each individual camera dependent on each other. As an example, the visual coverage of a single sensor in a coverage-maximization system could have different degrees of quality, depending if other cameras are monitoring the same areas or not. The local data must thus be analyzed in order to compute the global state, quality and resource consumptions F, Q and R. This requires a distribution—and in some cases an aggregation and fusion—of local information in order to compute a global state of the system. This can be accomplished either by a centralized approach, where all the data is transferred to a central node which is in charge of computing the global state, or a distributed approach. In the second case, all the nodes converge to a shared estimate of F, Q and R only through local communications between neighboring cameras.

c) Reconfiguration: There are various methods available for solving the reconfiguration problem. These methods can be distinguished based on the used algorithms, the degree of decentralization and the rate of execution. In general, reconfiguration is performed towards achieving a specific goal of the overall application. In the context of the proposed framework such objective is expressed in terms of quality maximization and/or resource consumption minimization.

III. State of the Art

In the past, many works have been published on active vision, especially in the field of robotics (see for example the survey by Chen et al. [4]). Here "active" means that the camera sensor is not simply acquiring data in a passive way, but actively reconfigures its own internal and/or external parameters (e.g., position, orientation, focus, and exposure) in order to optimize a specific task. More recently, several works investigated the extension of the active approach from a single sensor to networks of sensors. In this approach, the entire network is dynamically reconfigured in order to increase the system capabilities of acquiring useful information from the surrounding environment. Camera network reconfiguration is not just an application of standard active vision techniques to multiple cameras, as the sensors are not independent from each other. New problems arise especially in terms of sensor coordination and distribution of computation (for an example of some typical problems to be faced in camera network reconfiguration systems see [5]).

Several works on network reconfiguration are based on wireless sensor networks, where sensors are typically assumed to be omnidirectional (e.g. temperature sensors) and the wireless communication infrastructure eases the reconfiguration tasks in terms of sensor (re-)deployment. Akyildiz et al. [6] present a detailed survey on wireless sensor networks. Visual sensors however imply stricter constraints on their capabilities compared to other sensors, in particular the directionality.

The rest of this section describes the most relevant works on camera network reconfiguration, organized by configuration space complexity. In section III-A we focus on resource-aware methods, in which reconfiguration is applied to those camera parameters that affect the resource consumption of the sensor, e.g., in terms of power consumption or bandwidth usage. These parameters typically include, but are not limited to, frame rate, image size, and in general any parameter involving image formation and transmission. These works often consider static cameras only, even though they can be extended to other types of sensors. In the next sections, we discuss more complex configuration spaces in which camera orientation and zoom are taken in consideration as well. Since the majority of the analyzed works fall in this case, we split them into two sections, representing the two main approaches developed so far. First, section III-B presents the coverage-oriented works, in which PTZ (pan-tilt-zoom) reconfiguration aims to achieve an optimal scene coverage. Second, section III-C discusses target-driven PTZ reconfiguration, in the sense that the camera parameters are controlled to optimize the acquisition of specific targets possibly moving inside the monitored scene. Finally, in section III-D the configuration space is extended to include the position of mobile cameras, and discusses configuration methods in UAV-based camera networks.

Each work is classified in terms of the key elements identified in the framework proposed in section II. The results of this analysis are summarized in Table I, where we identify the local processing, the configuration space, the fusion approach, the reconfiguration goal and the adopted reconfiguration algorithm for each mentioned work.

A. Resource-aware methods

Resource-aware methods explicitly focus on the available and/or required resources on each device while achieving a feasible performance. Hoffmann [7] studies four different heuristics for energy-aware resource allocation to achieve a given goal. The reconfigurator has to ensure minimal energy consumption and timely completion of the task. Hoffmann states that none of the investigated heuristics is always best for all evaluated situations.

Similar, dynamic power management (DPM) allows the system to change the operation state of system components during runtime based on their workloads. This reduces power consumption of the system while providing satisfactory overall performance. To omit a priori knowledge, Khan and Rinner [8], [9] employ a multi-layer artificial neural network to estimate the workloads in the reconfigurator, and hence select the optimal timeouts for each component in a multi-camera traffic surveillance system. Their system's quality measurements are based on user-specified power and performance constraints.

In distributed camera networks, Karuppiah et al. [10] coordinate resources to track persons in pairs of cameras. They rely on a fault containment unit as their reconfigurator to reallocate cameras during runtime in case persons are occluded, lost, or triangulation of the tracked person fails. To select a proficient camera pair, they propose two different policies—one focusing on stationary factors such as overlap and quality of triangulation, the other one exploiting dynamic environment factors such as activity density and the related trajectory dynamics. Rinner et al. [57] analyze a centralized,

TABLE I

CLASSIFICATION OF THE STATE OF THE ART ACCORDING TO THE KEY COMPONENTS OF THE PROPOSED FRAMEWORK. LOCAL PROCESSING REFERS TO THE ANALYSIS PERFORMED BY A SINGLE CAMERA, ALTHOUGH THIS INFORMATION IS NOT ALWAYS AVAILABLE (N/A) IN THE ANALYZED WORKS. THE CONFIGURATION SPACE REFERS TO PARAMETERS WHICH ARE MODIFIED DURING OPERATION. THE FUSION CAN BE PERFORMED IN CENTRALIZED (C) OR DISTRIBUTED (D) WAY. THE LAST TWO COLUMNS SUMMARIZE THE OVERALL RECONFIGURATION GOAL AND THE ALGORITHM ADOPTED TO ACHIEVE IT.

PAPER	LOCAL PROCESSING	CONFIGURATION SPACE	FUSION APPROACH	RECONFIGURATION GOAL	RECONFIGURATION ALGORITHM
[7]	n/a	Resource allocation	C	Task completion & resource consumption	Heuristics
[8], [9]	Surveillance	State selection	C	Resource consumption	Reinforced learning & ML-ANN
[10]	Tracking	Camera selection	C	Tracking quality & resource consumption	Failure containment
[11]	Tracking	Camera Selection	D	Tracking & target image quality	Socio-economic approach
[12]	n/a	Camera selection & energy distribution	C	Coverage & resource consumption	Stochastic model
[13]–[15]	Detection & tracking	Camera selection, task assignment & PTZ	D	Coverage & resource consumption	Message passing & expectation-maximization
[16], [17]	n/a	Position & PTZ	C	Coverage, image quality	Simulated annealing
[18]	Detection	PTZ	D	Coverage, resolution	Incremental line search
[19]	n/a	Position & orientation	C	Coverage	Particle swarm optimization
[20]	n/a	Position & orientation	C & D	Coverage	Integer linear programming & greedy
[21]	n/a	PTZ	C	Coverage & image quality	Particle swarm optimization
[22]	n/a	Position & orientation	C	Coverage	Greedy min-set cover
[23]	n/a	Position & PTZ	C	Coverage	Genetic algorithm
[24]	Task-specific	Position & orientation	C	Task-oriented coverage	Hill-climbing
[25], [26]	n/a	PTZ	C	Coverage	Expectation-maximization
[27]	Detection	Position & orientation	D	Target visibility	BIP, greedy
[28]	Detection & tracking	PTZ	C	Target image quality	sensor slaving
[29], [30]	Tracking	PTZ	C	Tracking	POMDP
[31]	Tracking	Camera selection	C	Tracking	Game-theoretic
[32]–[35]	Tracking	Camera selection & PTZ	D	Tracking & target image quality	Game-theoretic
[36], [37]	Tracking	Camera selection & PTZ	C	Tracking	Greedy best-first search
[38]	Tracking	Camera selection & PTZ	C	Tracking	Production rules
[39]	Detection	Position	C	Detection quality	Probabilistic occupancy map
[40]–[42]	Detection	Position	D	Coverage	Greedy search algorithm
[43]	Detection	Position & PTZ	C	Path planning	Receding-horizon optimization
[44], [45]	Feature extraction	Position	D	Path planning	Optimization
[46], [47]	Detection / Tracking	Position	D	Tracking	Task assignment
[48]	n/a	Position & PTZ	D	Coverage	Optimization
[49]–[51]	n/a	Position	C	Coverage	Coverage path planning
[52]	n/a	Position	D	Coverage	Max-sum task assignment
[53]	Identification	Position	D	Tracking	Negotiation
[54], [55]	Path selection	Position & direction	D	Tracking	Optimization
[56]	Detection	Position	D	Coverage	Lawn mower search pattern

distributed and proprioceptive approach to assign tracking responsibilities in smart camera networks. Their analysis focuses on resource consumption in terms of exchanged messages as a quality metric. This communication overhead as a proxy for computational power, as cameras only process images when they are requested to do so. Yu and Sharma [12] define a stochastic model of the operational lifetime of a visual sensor network. They consider two problems for their reconfigurator: (i) the selection of a set of cameras to achieve a desired coverage and (ii) the distribution of energy among all selected nodes. Based on an abstraction of their proposed model, they are able to optimize the lifetime of the network. Dieber et al. [13] propose a distributed algorithm to find a network wide configuration with a focus on a good trade-off within their quality metrics: surveillance quality and resource consumption. They compare their distributed approach to a centralized, evolutionary approximation and show that they can achieve equal configuration qualities. In [14] they extend their approximation method with an expectation-maximization algorithm to optimize coverage and resource allocation in distributed camera networks including PTZ cameras. To deal with even higher dynamics, Dieber et al. [15] merge the distributed reconfiguration approach with the market-based assignment of tracking responsibilities. This represents a particular implementation of our presented generalized framework.

B. Coverage-oriented methods

Coverage-oriented methods aim to optimize the global coverage of the monitored scene, e.g., by maximizing the number of sensors observing relevant areas or minimizing the unobserved portions of the environment. The problem has been initially formulated and studied in the field of computational geometry, where it is known as the "art gallery problem" [58]. The original problem is proven to be NP-Hard, but several approximate solutions and variants have been proposed (for an example, see [59]). The original art gallery problem however cannot be directly applied to real distributed camera networks, as it does not consider several issues such as camera directionality or range. Surveys on coverage modeling for real cameras include Mavrinac and Chen [60], where different geometrical and topological models for camera coverage are analyzed, Guvesan and Yavuz [61], which focuses on networks of directional sensors and their similarities/dissimilarities with wireless sensor networks, and Zhao et al. [62].

One of the most relevant works on coverage optimization has been proposed by Mittal and Davis [16], [17]. The authors assume that the probability distribution of random objects appearing in the scene, as well as object characteristics such as appearance and geometry, are known. By combining this information with prior knowledge on the static structure of the scene, they propose a probabilistic framework modeling the scene visibility given the presence of random occlusions due to dynamic objects. The authors also combine this model with image capture quality metrics to define a global quality function. The reconfigurator uses a centralized approach based on simulated annealing to optimize this function and finds the configuration that maximizes the scene visibility and image quality even in presence of random occlusions. The configuration space consists in sensors position, orientation and zoom, as well as acquisition parameters related to image quality, such as image resolution.

In Kansal et al. [18] the configuration space consists in the pan, tilt and zoom values of fixed-position PTZ cameras. Their system collects information on scene topology by using laser ranging devices, and learns the zones of high activity by detecting moving objects in wide-view, low-resolution image sequences. A quality metric is defined as a function of several parameters such as scene coverage (with higher importance given to high-activity zones), object visibility and resolution, and actuation delay. The authors propose a distributed approach in which each camera computes its local quality metrics and is able to communicate only with other sensors in a limited neighborhood. Incremental Line Search is used as a reconfiguration algorithm to let each camera converge to a shared global performance-maximizing configuration. The system can dynamically reconfigure and adapt to changes in activity maps.

The work by Morsly et al. [19] deals with the initial deployment of static cameras: the configuration space thus consists in the sensors positions and orientations. The authors model the sensor capabilities as constraints for an optimization problem that minimizes the required resources, defined as the number of sensors needed to guarantee that each point in the scene is sensed by at least one camera. The reconfigurator is based on a probability-inspired binary particle swarm optimization technique. The proposed system uses a centralized approach.

The work of Ai and Abouzeid [20] focuses on the coverage of a discrete set of targets with generic directional sensors. The configuration space consists in both position and orientation of the sensors, and the goal is to maximize a global quality function defined as the coverage of the targets, while minimizing the required resources (number of sensors). The problem is solved both exactly with an integer linear programming approach, and approximately with a computationally efficient greedy algorithm. The greedy algorithm is formulated both in a centralized and a distributed version.

Konda and Conci [21] propose an multi-purpose approach to PTZ configuration. The configuration space consists in both the orientation and zoom levels of each sensor. The authors aim to maximize two quality measures, camera network coverage and image quality, defined in terms of image entropy and distortion. The reconfigurator optimizes these metrics using a centralized particle swarm optimization approach.

Angella et al. [22] discretize the observation and the camera placement spaces, and define a proper inter-visibility measure between points in 3D environments. The visibility measure is used to define a global coverage quality score, whose maximization is found by applying a variant of the centralized greedy algorithm for min-set cover problems. They also propose a software implementation which can be hardware-accelerated by using modern graphic cards. The configuration space consists in both the sensor positions and orientations.

Also Indu et al. [23] discretize the environment space. They also require manual definition of special areas such as the zones where camera placement is possible and priority areas where sensor coverage is mandatory. They define a probability-based quality measure related to scene coverage. The configuration space considers the orientation and zoom levels of the cameras, as well as their position in the case of patrolling PTZ cameras. The reconfigurator finds the optimal coverage by means of a centralized genetic algorithm.

Bodor et al. [24] define reconfiguration goal in terms of maximizing a quality measure expressing the coverage of areas where most of the activities are taking place. Moreover, they observe that different tasks have different needs in terms of image quality, and thus they propose a task-oriented optimal network configuration based on a hill-climbing centralized algorithm.

Piciarelli et al. [25], [26] propose a method for the automatic reconfiguration of a network of PTZ cameras based on the notion of relevance maps. A relevance map is a 2D (in [25]) or 3D (in [26]) map expressing the relevance for each visible portion of the environment based on the goals of the system. Its definition is thus highly dependent on the global goal: a typical example could be an activity map expressing the presence of moving objects detected by wide-FOV static cameras or noisy targets localized by audio sensors [63]. In this case, the authors assume that the zones with higher activity are also the most relevant for surveillance. The final aim of the system is to compute a reconfiguration of the pan, tilt and zoom parameters of each camera in order to maximize the global coverage, giving more weight to high-relevance zones. The authors propose a local coverage model that expresses, for each camera, the quality of the current local configuration, defined as the total amount of relevance of the zones observed by the sensor. The Fusion module acts in a centralized way and computes the global quality of the current configuration as a sum of the local qualities. The final goal of the system is to maximize the global quality, and this is obtained by the reconfigurator module by iteratively computing the best network configuration using an approach based on the Expectation-Maximization algorithm. The configuration space consists in the pan, tilt and FOV angles of each camera.

C. Target-based methods

In contrast to coverage-oriented approaches, which aim to maximize the area coverage independently of the presence of objects in the scene, target-based methods use reconfiguration to focus on a specific target. Target tracking is the most typical application, and can be accomplished by tasking PTZ cameras to actively follow the tracked objects while other cameras adapt their configuration to observe the remaining environment. Reconfiguration can occur even with static cameras, and the network reconfiguration consists in actively selecting the best cameras to acquire a specific target (camera assignment and hand-off problems).

Zhao et al. [27] deals with the problem of detecting specific visual features (tags) on targets, e.g., faces, even in case of occlusion. They propose a stochastic visibility model to guarantee that tags are visible from at least two cameras, thus allowing the distributed 3D localization of the tags also in occluded views. Then, they introduce two algorithms for their reconfigurator to compute the camera positions and orientations that maximize the tag visibility. Finally, they present a practical privacy-oriented application that removes the presence of a tagged person from all the camera views. Del Bimbo et al. [28] reconfigure the network to acquire high-resolution images of a target with PTZ cameras. They rely on a master-slave approach, in which a master camera monitoring a large environment provides rough positional information to slave cameras. Each camera in the network can act either as master or slave. Their reconfigurator uses an uncalibrated method to estimate the time-variant homographies between camera views. Natarajan et al. [29], [30] try to overcome the limits of a master-slave approach and propose a model to coordinate a network of PTZ cameras for tracking multiple targets with high resolution even in presence of occlusions. In their approach, the reconfigurator maintains a belief on the states of the targets which is considered in the framework of Partially Observable Markov Decision Processes. Qureshi and Terzopoulos [36], [37] consider the problem to track moving objects with PTZ cameras while optimizing the image quality and minimizing the number of camera hand-offs. The novelty of their approach consists in considering the predicted trajectories of moving objects in order to optimize the camera tasking strategies through time for future situations. A proactive approach is proposed also by Starzyk & Qureshi [38], where a reasoning module tries to find the best camera assignments and hand-offs, and reuses the same configuration when similar events occur. The authors also extend their works to the case of multiple tasks executed by each camera simultaneously [64].

Regarding camera assignment and hand-off systems, Li and Bhanu [31] define multiple utility functions as a quality measure to analyze the system efficiency in target tracking from the point of view of each sensor. Then, a global optimal camera assignment is found by using the utilities in a game theoretic framework, where the global utility is maximized by a bargaining process on a centralized reconfigurator. The proposed system works both with overlapping and non-overlapping cameras and does not require specific offline camera calibration procedures. Results are given on people tracking and face detection. The same authors compare several camera selection and hand-off techniques in [65]. A similar approach is used by Song et al. [32] and by Ding et al. [35], where again utility functions are proposed to model the tracking quality of moving people and the optimal camera selection is found by means of a game-theoretic approach. However, in these works the system is fully decentralized, and PTZ configuration is considered for acquiring high-resolution images of specific targets. The same authors in [33] improve the system tracking performances by using a distributed Consensus-Based Kalman filter, and propose a more generic framework based on their approach in [34], [66]. Esterle et al. [11] present a completely distributed coordination approach exploiting market mechanisms. Several strategies are proposed to use a local utility function combined with distributed Vickrey auctions to assign tracking responsibilities during runtime to the ”best” camera. As no a priori knowledge is facilitated, the cameras need to learn the network topology online and use artificial pheromones to represent the local neighborhood. The reconfigurator selects appropriate neighbors to focus marketing efforts. In [67] they show the robustness of their approach in uncertain environments but define the employed strategies at deployment time. To overcome an initial—and possibly non-optimal—assignment of strategies, Lewis et al. [68] propose to facilitate so-called bandit solvers on each camera to learn the best strategy during runtime. Here the focus of the reconfigurator shifts from selecting appropriate cameras as recipients of auction invitations to selecting the most appropriate strategy locally on each camera. Starzyk and Qureshi [69] adopted the approach by Esterle et al. and applied one of the strategies for coordinating the handover in a network of PTZ cameras. Additionally, they allowed cameras to share their current state with others and to perform so-called counteroffers, where the bidding camera offers one of its objects for sale in order to free allocated resources.

In a collaborative tracking approach SanMiguel and Cavallaro [70] use their reconfigurator to select a coalition manager for each object of interest based on available battery levels and current load. The coalition manager is dynamically changed during runtime based on the objects location. The first task of the coalition manager is to identify the trade-off between the contributions of each camera tracking the object against the effort to include it in the coalition for this specific object. The second task is to estimate the target state over time based on the information of the cameras within the coalition.

The problem of camera assignment has also been addressed by Martinel et al. [71] in the context of human-computer interfaces. In their work trajectory prediction of moving objects is used to dynamically select the best cameras from the configuration space observing a specific target. This camera selection allows to reduce the bandwidth consumption since only the best video streams are sent to a centralized node for human operator evaluation.

D. UAV-based deployment methods

Recent developments in UAVs [72], flying ad-hoc networks [73], and computer vision have enabled the deployment of UAVs for mobile camera networks. Each node in such a network is a position-aware UAV, carrying a camera (static or PTZ) for vision-based sensing. A real time deployment strategy is always required to decide on the configuration of nodes (i.e., position and movement action of a UAV) in order to achieve the mission goals. Reconfiguration in this kind of network is based on the concept of minimizing uncertainty in the environment or maximizing information gain about the environment. One natural application of a UAV-based camera deployment is situation-awareness in wide areas, which includes detection, search and rescue, tracking, surveillance and monitoring. Situation-awareness in natural environments generally requires high mobility and perception capabilities of mobile nodes.

Merino et al. [39] propose a cooperative perception system for automatic forest fire detection and precise localization using camera-equipped UAV network. In their proposed method, each UAV locally uses segmentation, and feature matching to find the estimates of fire locations. These local estimates (quality measures) are then fused using Bayesian analysis to further increase the accuracy of the fire locations. The reconfigurator uses the centralized occupancy probability map and unscented transform to position the UAVs for reducing the errors in localization. Similar camera-equipped UAV networks have been used in various search and rescue scenarios (e.g., [40]–[45]), where computer vision algorithms are used to calculate the probability of detection and probability of false alarm for the vision sensors. Considering these probabilities as quality measure, the observations from UAVs are fused to localize a stationary target. Recently, Khan et al. [56] introduced a generic framework of aerial mobile camera network for search applications, where the local analysis is to update the probability of target existence based on individual UAV observations. The authors proposed and compared various strategies for merging the time delayed and erroneous observations from individual UAVs. The estimated probability of target existence is then used as a confidence of the individual UAV in the target location. The positions of UAVs along the predefined paths are then reconfigured in order to minimize the search time and detection errors.

How et al. [46] and Bethke et al. [47] propose system architectures for distributed search and track for a moving target. To provide accurate target state estimation, even in the presence of obstructions in the environment, they use optimization technique to combine instantaneous observations from individual UAVs. This optimization is based on minimizing the errors in distance from the last known estimate to each measurement. The target state estimation is then used by linear Kalman filter to determine the next location for observation. A similar approach is used in [53] to monitor the activities of firemen in disaster situation. The reconfiguration in their method is based on a negotiation process. The UAVs negotiate to decide the future locations based on minimization of Euclidean distances to firemen. To track multiple moving targets, the authors in [54], [55] claimed that uncertainty of a target state is proportional to frequency of target state update. They formulated the motion planning problem of UAVs as an optimization problem to minimize the average time duration between two consecutive observations of a target. A gradient-approximation algorithm is then proposed as a reconfigurator to generate suboptimal paths for the UAVs to traverse.

One of the most important works in UAV-based deployment for camera network reconfiguration is environmental monitoring, proposed in [48]. The authors propose a method to coordinate heterogeneous cameras and to maintain the best view of the environment with maximal resolution. They derive a cost function to represent how well a group of cameras covers the environment. They use information per pixel as a metric to incorporate physical, geometric, and optical parameters into this cost function. A control law is then obtained for reconfiguring the positions and orientations of camera by taking the negative gradient of this cost function. The authors in [49]–[51] propose techniques for generating an overview image of a large scene. Images are received at a fixed base station from multiple nodes that are planned to travel along specific waypoints (configuration space). These waypoints are determined in a static manner but the dynamic calculation of efficient paths for UAVs are formulated as integer linear programming problem. Image registration and mosaicking are then applied to form an overview image of the defined area. Fave et al. [52] provide refined imagery of multiple scenes prioritized by the user. Based on the priorities, the shortest distance to scene, and the available battery life, tasks are allocated to UAVs. The reconfigurator uses dynamic task allocation to continuously change the positions and paths of UAVs.

IV. Reconfiguration challenges

From the analysis presented in section III, it emerges that network reconfiguration is a complex task that cannot be reduced to the mere application of standard optimization techniques. Here we summarize the main challenges that we have identified in the analyzed works, and that are specific to camera network reconfiguration problems:

- *Complexity of multiple goals and constraints.* The proposed framework aims to maximize a global quality measure for sensing capabilities of the network, based on the local measurements of each individual sensor. While the definition of the quality measure sometimes comes naturally from the problem definition (e.g. the total size of the observed area in coverage-oriented tasks), in can be the most challenging problem in other cases. A typical example comes from object tracking tasks: how can we decide if a network configuration is good for tracking? The answer depends on the specific context, for example Zhao et al. [27] require each object to be observed by at least two cameras, Del Bimbo et al. [28] focus on acquiring high-resolution images of the targets, while Qureshi and Terzopoulos. [37] minimize the number of camera hand-offs. Finding a good quality measure and defining how it can be computed by only using local measurements of each sensor is thus a though task, tightly coupled with the global goal of the system.
  Moreover, there is a clear trend towards deployment of camera networks in complex environments which consequently requires more complex configuration objectives. Here the reconfiguration has to fulfill multiple, sometimes even conflicting, criteria. In such settings, a set of (Pareto-optimal) configurations are available, and knowledge about the current context helps to select a configuration. Thus, context-awareness or situation-awareness is desirable for reconfigurations in complex environments.
  Finally, network reconfiguration systems need to deal with multiple complex constraints. These constraints arise either from the physical architecture of the sensors, such as pan/tilt ranges, zoom levels, power requirements; or from the environment, such as visual occlusions.
- *Dynamic reconfiguration and real-time processing.* In some cases, the optimal network configuration is an offline process that leads to a static solution. This is especially true for many coverage-oriented problems, where the environment is considered static and it is assumed that the coverage of each sensor will not change through time. However, in many applications the network reconfiguration is a continuous process, since the network must dynamically adapt to the changing conditions of the monitored environment. A good example is target tracking: at each time instant, the network must adapt its configuration to the position of the moving objects being tracked. This dynamic aspect is often ignored by many standard optimization techniques.
  The dynamic nature of reconfiguration also implies stringent requirements in terms of computational time. Especially in active camera networks, real-time processing is required in order to allow the reconfiguration operation to keep up with the dynamics of the environment. The limited resources available within the camera network pose therefore a fundamental challenge for dynamic reconfiguration. Offloading the computation to powerful servers or even to cloud services—a technique which is frequently applied in traditional camera networks and computer vision applications to overcome resource limitations—is often not feasible. Dynamic reconfiguration has therefore proactively monitor the available resources and aim for computing configurations within guaranteed time bounds. Heuristics, incremental algorithms or approximation methods might be potential algorithm candidates.
- *Distributed computation.* As shown in Table I, the majority of the surveyed works still adopts a centralized approach to information distribution and fusion. In this case the network relies on a central processing node connected to all the sensors that computes the required reconfiguration. While this approach is suitable for small networks, it does not scale for larger installations. We foresee that camera networks will continue to grow in the near future—even to the scale of metropolitan areas. Thus, distributed computation will become a necessity, and the use of smart sensors, with embedded processing capabilities, will be a key factor to implement fully-distributed reconfiguration algorithms.
- *Increased system uncertainty.* Advanced camera networks have to deal with uncertainties among various dimensions. First, knowledge about the current state of the camera network may be limited due to resource limitations and failures. Second, knowledge about the available configurations of the network may also be limited. Third, sensing and low-level processing may only provide incomplete information about the environment due to lack of observation and detection errors, for example. Finally, the environment may also provide unexpected behavior which was not anticipated at time of deployment. One fundamental approach to address the uncertainty challenge is to exploit online or reinforcement learning strategies.
- *Self-organizing networks.* The ultimate vision of reconfiguration is self-organization where the camera network autonomously monitors its state and progress, learns behaviors and adapts itself to reflect changing conditions of the environment. Learning models of the camera's state and context as well as decentralized decision making are the

fundamental techniques for achieving self-organization. An extension to self-organization is self-awareness and self-expression. By introducing self-awareness and self-expression, cameras will not only be able to learn about their own state, but also to reason about their capabilities. Having a plethora of capabilities, a self-aware camera is able to do both: to decide *when* to apply a certain technique or mechanism as well as to decide *what* the most appropriate available method is during runtime.

V. Discussion

In this section we discuss the key aspects that emerge from an overall comparison of the works presented in section III. In particular we focus on three main elements shared by any camera reconfiguration work, namely:

- reconfiguration goals
- configuration spaces
- reconfiguration strategies

The outcome of this analysis is summarized in table II. The table is meant as a quick reference to identify relevant works given a specific applicative scenario.

Reconfiguration goals. Current state-of-the-art works are focused on the optimization of three main aspects: the visual coverage, the tracking performances, and the overall resource consumption of the sensing network. Visual coverage consists in maximizing the total area observed by the cameras. This has been generally done by using some a-priori knowledge on the static structure of the environment to compute its visibility given a network configuration. None of the analyzed works considers the possibility of dynamic occlusions, with the notable exception of [16], [17], in which however the random occlusions caused by dynamic objects are handled only from a probabilistic point of view.

The work by Kansal et al. [18] extends the basic idea by introducing weights, in order to give more importance to the coverage of relevant zones. Most of the coverage-oriented works are not focused on the dynamic aspects of reconfiguration, and the proposed techniques are rather used to find an static optimal configuration (e.g., after an sub-optimal initial deployment of the sensors). Exceptions can be found in weighted coverage works, in which the weights could change. Piciarelli et al. [25], [26] compute the weights by detecting the presence of moving objects, and thus these weights can dynamically evolve through time.

The goal of tracking-oriented works, on the other hand, are not so well defined. All of the works aim to improve the network-wide performance of object tracking or some other target-based processing, however there is no general consensus on how this goal can be achieved. The aim of the system is actually very application-specific: in [27] the authors want some visual features (e.g. faces) to be always visible by at least a camera, in [28] sensors are reconfigured to acquire high-resolution images of the targets, while the work proposed in [21] optimizes a metric modeling the overall image quality. In general, all these systems assume that tracking will benefit from an improved image quality. Another common approach to improve tracking is to minimize the camera hand-offs (changing the sensor in charge of tracking a specific target), such as in [11], [31], [34], [66].

This leads directly to the third main goal, minimizing resource consumption of each individual camera and hence the entire network. The rationale behind minimizing hand-offs is that vision-based tracking gives better results on long sequences from the same sensor, moreover each camera hand-off requires a non-trivial re-identification of the lost target in the other views. Several works try to find a trade-off between the two possibly conflicting goals, trying to optimize the image quality while minimizing the hand-offs at the same time, such as in [36], [37]. Lewis et al. [68] even use different strategies in order to determine the Pareto-efficient frontier when trading off tracking performance against required resources. Furthermore, they use reinforcement learning techniques in combination with weighted utility function to learn Pareto-optimal solutions for their available strategies during runtime.

Configuration spaces. Another key aspect to consider in network reconfiguration is the configuration space, this is, the set of sensor parameters that can be altered and that influence the network performance in some way. The configuration space depends on the sensor mobility, i.e. if the sensors are static, PTZ or fully mobile, and this distinction has been used to organize section III. In static cameras, only the internal parameters can be modified, such as optical aperture, resolution, acquisition rate etc. (see section III-A). These parameters affect the image quality and the power consumption, and thus they are often reconfigured in order to find a quality/resource trade-off in a resource-aware context, such as in [13]. Sometimes network reconfiguration simply reduces to turning specific sensors on or off, as in [74]: while aiming to achieve a desired sensing performances while using the sensors as little as possible, thus reducing resource consumption. In general, all the analyzed works that rely on static cameras are oriented to resource awareness. Nevertheless, resource awareness is achieved in several works (e.g. [14]) using PTZ sensor networks as well.

Pan-Tilt-Zoom (PTZ) cameras offer a more complex configuration space, since the camera orientation becomes a configurable parameter. The zoom parameter, when configured, is always handled together with pan/tilt angles: in our analysis, no work controlled the zoom in static cameras. The majority of network reconfiguration works rely on PTZ architectures, as shown in section III. The main problems addressed with this type of sensors are visual coverage (section III-B) and target tracking (section III-C). Indeed, given the limited field-of-view of most of the cameras, controlling the sensor orientation is an effective way to achieve better visual coverage or to keep with a moving target in the observable area.

Finally, the configuration space can be further extended in order to model not only the sensor orientation, but also its position. In this case the reconfiguration strategy can take advantage of the full mobility of the sensors, which can be dynamically deployed according to the sensing needs. While this approach can be theoretically applied to any type of mobile sensor, the totality of the works we analyzed involve the usage of unmanned aerial vehicles (UAVs). UAV-based systems are typically used to deal with both the visual coverage and target tracking in large geographical regions. Limitations in battery life (flight time), field-of-view, and computing power of the UAVs require intelligent and dynamic positioning of UAVs to visually cover the region or to track the targets in the region. Despite current reconfiguration works with mobile sensors focus on UAV platforms, we foresee that in the near future this approach could be extended to other types of sensors as well, such as wearable equipment, tablets or smartphones.

Reconfiguration strategies. Once the reconfiguration goals and parameters have been identified, a proper reconfiguration strategy must be chosen. There is no general consensus on which algorithms suit this task best, but all the analyzed works can be expressed in form of nonlinear optimization problems in the end, and have been handled accordingly with tools like genetic algorithms [23], simulated annealing [17], particle swarms [19], etc.

The choice of a proper algorithm also depends on the specific challenges that the system must face, as the ones described in section IV. In particular, the nature of the problem leads naturally to either a centralized or distributed approach. In the centralized approach, all the computation is required to be accomplished by a single central computing node. This node knows the global state of the system and can compute the best configuration for each sensor on the network, however it is also a single point of failure, and can be a computational bottleneck in large networks. Distributed systems, on the other hand, rely on distributing the computation on several smart sensors, which mutually agree on a shared global reconfiguration strategy by exchanging messages among the sensors. The distributed approach is generally harder to implement, however it performs better in terms of scalability, and it is typically able to deal with the failure of one or more sensors. From table I we can see that the number of works using the different approaches is roughly the same, however there is a clear trend towards the centralized or distributed strategy depending on the reconfiguration goal. In particular, tracking oriented systems tend to use a centralized approach [28]–[30], [36]–[38]. We speculate that this is due to the limited area to be monitored and the low number of employed sensors in the network. In this case, the computational burden is not critical, and the authors prefer a centralized strategy, which is generally easier to implement. On the other hand, the distributed approach is widely adopted in the UAV-based works [42], [53]–[56]. This is often due to limitations in the wireless communication [75]: while it could be feasible to have local communications between UAVs, the wireless range could limit the communication with a single remote node.

The other fundamental challenge is to deal with dynamic problems, possibly requiring a real-time reconfiguration. From this point of view, the two extrema are represented by coverage-oriented and tracking-oriented problems. In the first case the reconfiguration is often computed only once, using static information on the structure of the environment, and thus can be safely performed offline. No dynamic adaptation is needed and the computational time is not a critical issue. Undeniably, there are situation where the importance of covering a certain area may change during runtime (cp. [26]) and an online reconfiguration is inevitable for coverage-oriented problems. On the other hand, tracking-oriented approaches always need to deal with extremely dynamic contexts, since their reconfiguration depends on the movement of the tracked objects. In this case, a real-time reaction is critical, since a delay in reconfiguration could lead to the loss of the target. For this reason, tracking-oriented works often try to find fast, possibly approximate solutions, for example using greedy algorithms as in [27], [36], [37].

VI. Conclusions

In this work, we have presented two main contributions. Our first contribution proposes a novel framework for reconfiguring smart camera networks. This framework enables several cameras to work together towards a common goal under global quality and resource constraints. Our second contribution is a short survey of some of the most relevant state-of-the-art works, allowing us to identify benefits as well as drawbacks in current systems. The state of the art has been analyzed in the novel context of the proposed framework. As a result of this analysis, we have identified the main challenges to be faced in the context of network reconfiguration, and we concluded with a discussion of the most suitable reconfiguration strategies in terms of the reconfiguration goals and parameter spaces.

Acknowledgments

This work is partially supported by the Italian Ministry of Defense project PRNM n. 9900 “ADVISOR III”, by the EU FP7 program under grant agreement n^{o} 257906 `http://www.epics-project.eu/`, and by Lakeside Labs GmbH, Klagenfurt Austria with funding from the European Regional Development Fund and the Carinthian Economic Promotion Fund (KWF) under grant 20214/21530/32602.

References

[1] B. Rinner and W. Wolf, “Introduction to Distributed Smart Cameras,” *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1565–1575, 2008.

[2] M. Reisslein, B. Rinner, and A. Roy-Chowdhury, “Smart Camera Networks (Guest Editors’ Introduction),” *Computer*, vol. 47, no. 5, 2014.

[3] J. C. SanMiguel, K. Shoop, A. Cavallaro, C. Micheloni, and G. L. Foresti, “Self-Reconfigurable Smart Camera Networks,” *Computer*, vol. 47, no. 6, pp. 67–73, 2014.

[4] S. Chen, Y. Li, and N. M. Kwok, “Active Vision in Robotic Systems: A Survey of Recent Developments,” *The International Journal of Robotics Research*, vol. 30, no. 11, pp. 1343–1377, 2011.

[5] C. Micheloni, B. Rinner, and G. L. Foresti, “Video Analysis in Pan-Tilt-Zoom Camera Networks,” *Signal Processing Magazine*, vol. 27, no. 5, pp. 78–90, 2010.

[6] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless Sensor Networks: A Survey,” *Computer Networks*, vol. 38, pp. 393–422, 2002.

[7] H. Hoffmann, “Racing and Pacing to Idle: An Evaluation of Heuristics for Energy-aware Resource Allocation,” in *Proceedings of the Workshop on Power-Aware Computing and Systems*. ACM, 2013, pp. 1–5.

[8] U. Khan and B. Rinner, “A Reinforcement Learning Framework for Dynamic Power Management of a Portable, Multi-camera Traffic Monitoring System,” in *Proceedings of the International Conference on Green Computing and Communications*. IEEE Computer Society Press, 2012, pp. 557–564.

[9] ——, “Online Learning of Timeout Policies for Dynamic Power Management,” *Transactions on Embedded Computing Systems*, vol. 14, pp. 1–25, 2014.

TABLE II
SUMMARY OF THE ANALYZED WORKS, CLASSIFIED ACCORDING TO THE MAIN GOAL (COVERAGE, TRACKING, RESOURCE AWARENESS) AND CONFIGURATION SPACE (INTERNAL PARAMETERS, PTZ, POSITION). THE TABLE ALSO CONTAINS CONSIDERATIONS ON RECONFIGURATION STRATEGIES (E.G. CENTRALIZED VS DISTRIBUTED) AND REAL TIME REQUIREMENTS.

	COVERAGE	TRACKING	RESOURCES
INTERNAL PARAMETERS	Coverage optimization cannot be achieved by reconfiguring only the internal camera parameters	Camera hand-off systems. See for example [31]. In case of large network, consider a distributed approach as in [11], [32]–[35], [69].	
PT(Z)	**Small Networks:** Scalability is not a issue: use a centralized approach. No stringent real-time requirements. See [16], [17], [19]–[24]. If the coverage can dynamically evolve through time, see [25], [26].	Typically centralized approaches for small networks. Real-time requirements lead to fast, possibly approximate solutions. See [28]–[30], [36]–[38]. For an example of distributed approach, see [27].	Distributed approach to maximize the trade-off between coverage and resource utilization in [14].
	Large Networks: Use a distributed approach to face scalability issues. No stringent real-time requirements. See [18], [20].		
POSITION	Communication with a fixed central node could be unfeasible: use a distributed approach. Computational load could be a limit for embedded systems, use fast approximate algorithms. See [40]–[42], [48], [52], [56]for distributed approaches. For an example of centralized approaches, see [49], [51].	Same considerations of coverage-oriented systems apply. Typically distributed approaches, as in [46], [47], [54], [55].	Partial resource awareness in terms of battery life. The systems are typically centralized, for example [49], [51].

[10] D. R. Karuppiah, R. A. Grupen, Z. Zhu, and A. R. Hanson, "Automatic Resource Allocation in a Distributed Camera Network," *Machine Vision and Applications*, vol. 21, no. 4, pp. 517–528, 2010.

[11] L. Esterle, P. R. Lewis, X. Yao, and B. Rinner, "Socio-economic Vision Graph Generation and Handover in Distributed Smart Camera Networks," *Transactions on Sensor Networks*, vol. 10, no. 2, pp. 20:1–20:24, 2014.

[12] C. Yu and G. Sharma, "Camera Scheduling and Energy Allocation for Lifetime Maximization in User-Centric Visual Sensor Networks," *Transactions on Image Processing*, vol. 19, no. 8, pp. 2042–2055, 2010.

[13] B. Dieber and B. Rinner, "Resource-aware Sensor Selection and Task Assignment," in *Proceedings of the International Conference on Advanced Video and Signal-Based Surveillance*. IEEE Computer Society Press, 2011, pp. 438–440.

[14] B. Dieber, C. Micheloni, and B. Rinner, "Resource-Aware Coverage and Task Assignment in Visual Sensor Networks," *Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 10, pp. 1424–1437, 2011.

[15] B. Dieber, L. Esterle, and B. Rinner, "Distributed Resource-aware Task Assignment for Complex Monitoring Scenarios in Visual Sensor Networks," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2012, pp. 1–6.

[16] A. Mittal and L. S. Davis, "Visibility Analysis and Sensor Planning in Dynamic Environments," in *Proceedings of the European Conference on Computer Vision*. Springer, 2004, pp. 175–189.

[17] ——, "A General Method for Sensor Planning in Multi-sensor Systems: Extension to Random Occlusion," *International Journal of Computing Vision*, vol. 76, pp. 31–52, 2008.

[18] A. Kansal, W. Kaiser, G. Pottie, M. Srivastava, and G. Sukhatme, "Reconfiguration Methods for Mobile Sensor Networks," *Transaction on Sensor Networks*, vol. 3, no. 4, pp. 1–28, 2007.

[19] Y. Morsly, N. Aouf, M. S. Djouadi, and M. Richardson, "Particle Swarm Optimization Inspired Probability Algorithm for Optimal Camera Network Placement," *Sensors Journal*, vol. 12, no. 5, pp. 1402–1412, 2012.

[20] J. Ai and A. A. Abouzeid, "Coverage by Directional Sensors in Randomly Deployed Wireless Sensor Networks," *Journal of Combinatorial Optimization*, vol. 11, no. 1, pp. 21–41, 2006.

[21] K. Konda and N. Conci, "Optimal Configuration of PTZ Camera Networks based on Visual Quality Assessment and Coverage Maximization," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2013, pp. 1–6.

[22] F. Angella, L. Reithler, and F. Gallesio, "Optimal Deploymont of Cameras for Video Surveillance Systems," in *Proceedings of the International Conference on Advanced Video and Signal based Surveillance*. IEEE Computer Society Press, 2007, pp. 388–392.

[23] S. Indu, S. Chaudhury, N. Mittal, and A. Bhattacharyya, "Optimal Sensor Placement for Surveillance of Large Spaces," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2009, pp. 1–8.

[24] R. Bodor, A. Drenner, P. Schrater, and N. Papanikolopoulos, "Optimal Camera Placement for Automated Surveillance Tasks," *Journal of Intelligent and Robotic Systems*, vol. 50, no. 3, pp. 257–295, 2007.

[25] C. Piciarelli, C. Micheloni, and G. Foresti, "Occlusion-aware Multiple Camera Reconfiguration," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2010, pp. 88–94.

[26] ——, "Automatic Reconfiguration of Video Sensor Networks for Optimal 3D Coverage," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2011, pp. 1–6.

[27] J. Zhao, S. Cheung, and T. Nguyen, "Optimal Camera Network Configurations for Visual Tagging," *Journal of Selected Topics in Signal Processing*, vol. 2, no. 4, pp. 464–479, 2008.

[28] A. D. Bimbo, F. Dini, G. Lisanti, and F. Pernici, "Exploiting Distinctive Visual Landmark Maps in Pan-Tilt-Zoom Camera Networks," *Computer Vision and Image Understanding*, vol. 114, no. 6, pp. 611–623, 2010.

[29] P. Natarajan, T. N. Hoang, K. H. Low, and M. Kankanhalli, "Decision-theoretic Coordination and Control for Active Multi-camera Surveillance in Uncertain, Partially Observable Environments," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2012, pp. 1–6.

[30] P. Natarajan, T. N. Hoang, Y. Wong, K. H. Low, and M. Kankanhalli, "Scalable Decision-Theoretic Coordination and Control for Real-time Active Multi-Camera Surveillance," in *Proceedings of the International Conference on Distributed Smart Cameras*. ACM, 2014, pp. 115–120.

[31] Y. Li and B. Bhanu, "Utility-based Camera Assignment in a Video Network: A Game Theoretic Framework," *Sensors Journal*, vol. 11, no. 3, pp. 676–687, 2011.

[32] B. Song, C. Soto, A. K. Roy-Chowdhury, and J. Farrell, "Decentralized Camera Network Control using Game Theory," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE, 2008, pp. 1–8.

[33] C. Soto, B. Song, and A. Roy-Chowdhury, "Distributed Multi-Target Tracking In A Self-Configuring Camera Network," in *Proceedings of the*

International Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Press, 2009, pp. 1486–1493.

[34] C. Ding, B. Song, A. Morye, J. A. Farrell, and A. K. Roy-Chowdhury, "Collaborative Sensing in a Distributed PTZ Camera Network," *Transactions on Image Processing*, vol. 21, no. 7, pp. 3282–3295, 2012.

[35] C. Ding, A. A. Morye, J. A. Farrell, and A. K. Roy-Chowdhury, "Opportunistic Sensing in a Distributed PTZ Camera Network," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2012, pp. 1–6.

[36] F. Qureshi and D. Terzopoulos, "Planning Ahead for PTZ Camera Assignment and Handoff," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2009, pp. 1–8.

[37] F. Z. Qureshi and D. Terzopoulos, "Proactive PTZ Camera Control," in *Distributed Video Sensor Networks*. Springer, 2011, pp. 273–287.

[38] W. Starzyk and F. Z. Qureshi, "Learning Proactive Control Strategies for PTZ Cameras," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2011, pp. 1–6.

[39] L. Merino, F. Caballero, J. Dios, J. Ferruz, and A. Ollero, "A Cooperative Perception System for Multiple UAVs: Application to Automatic Detection of Forest Fires," *Journal of Field Robotics*, vol. 23, pp. 165–184, 2006.

[40] S. Waharte, N. Trigoni, and S. J. Julier, "Coordinated Search with a Swarm of UAVs," in *Proceedings of the Annual Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*. IEEE Computer Society Press, 2009, pp. 1–3.

[41] A. Symington, S. Waharte, S. Julier, and N. Trigoni, "Probabilistic Target Detection by Camera-equipped UAVs," in *Proceedings of the International Conference on Robotics and Automation*. IEEE Computer Society Press, 2010, pp. 4076–4081.

[42] S. Waharte and N. Trigoni, "Supporting Search and Rescue Operations with UAVs," in *Proceedings of International Symposium on Robots and Security*. IEEE Computer Society Press, 2010, pp. 142–147.

[43] J. R. Riehl, G. E. Collins, and J. ao P. Hespanha, "Cooperative Search by UAV Teams: A Model Predictive Approach using Dynamic Graphs," *Transactions on Aerospace and Electronic Systems*, vol. 47, pp. 2637–2656, 2011.

[44] J. Tisdale, Z. Kim, and J. K. Hedrick, "Autonomous UAV Path Planning and Estimation," *Robotics & Atomation Magazine*, vol. 16, no. 2, pp. 35–42, 2009.

[45] D. T. Cole, P. Thompson, A. H. Göktoğan, and S. Sukkarieh, "System Development and Demonstration of a Cooperative UAV Team for Mapping and Tracking," *The International Journal of Robotics Research*, vol. 29, pp. 1371–1399, 2010.

[46] J. P. How, C. Fraser, K. C. Kulling, L. F. Bertuccelli, O. Toupet, L. Brunet, A. Bachrach, and N. Roy, "Increasing Autonomy of UAVs," *Robotics & Automation Magazine*, pp. 43–51, 2009.

[47] B. Bethke, M. Valenti, and J. How, "Cooperative Vision Based Estimation and Tracking using Multiple UAVs," in *Lecture Notes in Control and Information Sciences*. Springer, 2007, pp. 179–189.

[48] M. Schwager, B. J. Julian, M. Angermann, and D. Rus, "Eyes in the Sky: Decentralized Control for the Deployment of Robotic Camera Networks," *Proceedings of the IEEE*, vol. 99, pp. 1541–1561, 2011.

[49] M. Quaritsch, K. Kruggl, D. Wischounig-Strucl, S. B. M. Shah, and B. Rinner, "Networked UAVs as Aerial Sensor Network for Disaster Management Applications," *Elektrotechnik und Informationstechnik, Special Issue on Wireless Sensor Networks*, pp. 56–63, 2010.

[50] S. Yahyanejad and B. Rinner, "A Fast and Mobile System for Registration of Low-altitude Visual and Thermal Aerial Images Using Multiple Small-scale UAVs," *Journal of Photogrammetry and Remote Sensing*, pp. 1–14, 2014.

[51] A. Barrientos, J. Colorado, J. del Cerro, A. Martinez, C. Rossi, D. Sanz, and J. Valente, "Aerial Remote Sensing in Agriculture: A Practical Approach to Area Coverage and Path Planning for Fleets of Mini Aerial Robots," *Journal of Field Robotics*, vol. 28, pp. 667–689, 2011.

[52] F. M. D. Fave, A. Rogers, Z. Xu, S. Sukkarieh, and N. R. Jennings, "Deploying the Max-sum Algorithm for Decentralised Coordination and Task Allocation of Unmanned Aerial Vehicles for live Aerial Imagery Collection," in *Proceedings of the International Conference on Robotics and Automation*. IEEE Computer Society Press, 2012, pp. 469–476.

[53] I. Maza, F. Caballero, J. Capitan, J. M. de Dios, and A. Ollero, "Firemen Monitoring with Multiple UAVs for Search and Rescue Missions," in *Proceedings of the International Workshop on Safety Security and Rescue Robotics*. IEEE Computer Society Press, 2010, pp. 1–6.

[54] Z. Tang and Ü. Özgüner, "Motion Planning for Multitarget Surveillance with Mobile Sensor Agents," *Transactions on Robotics*, vol. 21, no. 5, pp. 898–908, 2005.

[55] R. R. Pitre, X. R. LI, and R. Delbalzo, "UAV Route Planning for Joint Search and Track Missions – An Information-Value Approach," *Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2551–2565, 2012.

[56] A. Khan, E. Yanmaz, and B. Rinner, "Information Merging in Multi-UAV Cooperative Search," in *Proceedings of International Conference on Robotics and Automation*. IEEE Computer Society Press, 2014, pp. 3122–3129.

[57] B. Rinner, B. Dieber, L. Esterle, P. Lewis, and X. Yao, "Resource-aware Configuration in Smart Camera Networks," in *Proceedings of the International Conference on Computer Vision and Pattern Recognition Workshops*. IEEE Computer Society Press, 2012, pp. 58–65.

[58] J. O'Rourke, *Art Gallery Theorems and Algorithms*. Oxford University Press Oxford, 1987, vol. 1092.

[59] A. Bottino and A. Laurentini, "A Nearly Optimal Algorithm for Covering the Interior of an Art Gallery," *Pattern Recognition*, vol. 44, no. 5, pp. 1048–1056, 2011.

[60] A. Mavrinac and X. Chen, "Modeling Coverage in Camera Networks: A Survey," *International Journal of Computer Vision*, vol. 101, no. 1, pp. 205–226, 2013.

[61] M. Amac Guvensan and A. Gokhan Yavuz, "On Coverage Issues in Directional Sensor Networks: A Survey," *Ad Hoc Networks*, vol. 9, no. 7, pp. 1238–1255, 2011.

[62] J. Zhao, R. Yoshida, S.-c. S. Cheung, and D. Haws, "Approximate Techniques in Solving Optimal Camera Placement Problems," *International Journal of Distributed Sensor Networks*, vol. 2013, pp. 1–15, 2013.

[63] C. Piciarelli, S. Canazza, C. Micheloni, and G. L. Foresti, "A Network of Audio and Video Sensors for Monitoring Large Environments," in *Handbook of Soft Computing for Video Surveillance*. CRC Press, 2012, pp. 287–315.

[64] W. Starzyk and F. Z. Qureshi, "Multi-tasking Smart Cameras for Intelligent Video Surveillance Systems," in *Proceedings of the International Conference on Advanced Video and Signal-Based Surveillance*. IEEE Computer Society Press, 2011, pp. 154–159.

[65] Y. Li and B. Bhanu, "A Comparison of Techniques for Camera Selection and Hand-Off in a Video Network," in *Distributed Video Sensor Networks*. Springer, 2011, pp. 69–83.

[66] B. Song, C. Ding, A. T. Kamal, J. A. Farrell, and A. K. Roy-Chowdhury, "Distributed Camera Networks," *Signal Processing Magazine*, vol. 28, no. 3, pp. 20–31, 2011.

[67] L. Esterle, B. Rinner, P. R. Lewis, and X. Yao, "Improved Adaptivity and Robustness in Decentralised Multi-camera Networks," in *Proceedings of the International Conference on Distributed Smart Cameras*. IEEE Computer Society Press, 2012, pp. 1–6.

[68] P. R. Lewis, L. Esterle, A. Chandra, B. Rinner, and X. Yao, "Learning to be Different: Heterogeneity and Efficiency in Distributed Smart Camera Networks," in *Proceedings of the International Conference on Self-Adaptive and Self-Organizing Systems*. IEEE Computer Society Press, 2013, pp. 209–218.

[69] W. Starzyk and F. Qureshi, "A Negotiation Protocol with Conditional Offers for Camera Handoffs," in *Proceedings of the International Conference on Distributed Smart Cameras*. ACM, 2014, pp. 1–7.

[70] J. SanMiguel and A. Cavallaro, "Cost-aware Coalitions for Collaborative Tracking in Resource-constrained Camera Networks," *Sensors Journal*, no. 99, pp. 1–1, 2014.

[71] N. Martinel, C. Micheloni, C. Piciarelli, and G. L. Foresti, "Camera Selection for Adaptive Human-Computer Interface," *Transactions on Systems, Man and Cybernetics: Systems*, vol. 44, no. 5, pp. 653–664, 2014.

[72] F. Kendoul, "Survey of Advances in Guidance, Navigation, and Control of Unmanned Rotorcraft Systems," *Journal of Field Robotics*, vol. 29, pp. 315–378, 2012.

[73] I. Bekmezci, O. K. Sahingoz, and S. Temel, "Flying Ad-Hoc Networks (FANETs): A Survey," *Ad Hoc Networks*, vol. 11, pp. 1254–1270, 2013.

[74] W.-C. Feng, E. Kaiser, W. C. Feng, and M. L. Baillif, "Panoptes: Scalable Low-power Video Sensor Networking Technologies," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 1, no. 2, pp. 151–167, May 2005.

[75] T. Andre, K. Hummel, A. Schoellig, E. Yanmaz, M. Asadpour, C. Bettstetter, P. Grippa, H. Hellwagner, S. Sand, and S. Zhang, "Application-driven Design of Aerial Communication Networks," *IEEE Communication Magazine*, vol. 52, no. 5, pp. 129–137, 2014.

Claudio Piciarelli (Member, IEEE) received the M.Sc. and Ph.D. degrees in Computer Science from University of Udine, Italy, in 2003 and 2008 respectively. He is currently Assistant Professor in the same university. His main research interests include computer vision, artificial intelligence, pattern recognition, and machine learning, on which he published more than 40 articles on international journals and conferences. He is currently mainly involved in the use of active vision in surveillance systems and in smart sensor planning and deployment for large sensor networks. He worked on several national and European projects on topics such as traffic monitoring, airports surveillance and counter-terrorism systems.

Lukas Esterle received the M.Sc. degree in Computer Science and the Ph.D. degree with distinction in Information Technology from the Klagenfurt University in 2010 and 2014, respectively. During his masters degree he worked at Rether Networks Inc. in Stony Brook, NY. Starting in 2011 he worked towards his Ph.D. at the same university where he is a research associate in the European FP7 project 'Engineering Proprioception in Computing Systems (EPiCS)'. Additionally, he is also teaching various courses on the master level since 2012. His main research interests include computer vision, self-organization, self-adaptation and resource awareness. He wrote several papers for international journals and conferences. His current research is focused on distributed smart camera networks and autonomous tracking of objects and persons without central coordination.

Asif Khan received the B.Sc. degree in Computer Systems Engineering from University of Engineering and Technology Peshawar, Pakistan in 2006 and M.Sc. degree in Electronics and Communication Engineering from Myongji University, South Korea in 2009. Since 2012 he is working towards his Ph.D , funded by Erasmus Mundus Joint Doctorate in Interactive and Cognitive Environments (EMJD ICE), at Klagenfurt University, Austria and Queen Mary University of London, UK. His main research interests include multi-UAV systems, decentralized coordination in UAVs and cooperative search.

Bernhard Rinner (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in Telematics from Graz University of Technology, Austria in 1993 and 1996, respectively. He is full professor and chair of pervasive computing at Klagenfurt University. He held research positions with Graz University of Technology from 1993 to 2007 and with the Department of Computer Science, University of Texas at Austin, from 1998 to 1999. His current research interests include embedded computing, embedded video and computer vision, sensor networks and pervasive computing. Prof. Rinner has been co-founder and general chair of the ACM/IEEE International Conference on Distributed Smart Cameras and has served as chief editor of a special issue on this topic in the Proceedings of the IEEE and IEEE Computer.

Gian Luca Foresti (Senior Member, IEEE) received the M.Sc. in Electronic Engineering and the Ph.D. in Computer Science from University of Genoa, Italy, in 1990 and in 1994, respectively. Since 2000 he is Full Professor of Computer Science at the University of Udine. From 2009 to 2012 prof. Foresti has been Dean of the Faculty of Education Science and actually he is Deputy Director of the Department of Mathematics and Computer Science. He is also Director of the International Master Degree Course in Multimedia Communication and Information Technology. His main interests involve Computer Vision and Image Processing, Multisensor Data and Information Fusion, Pattern Recognition and Neural Networks. Prof. Foresti is author or co-author of more than 300 papers published in international journals and conferences and he has been co-editor of five books in the field of Multimedia, Video surveillance and Data Fusion. He has been Guest Editor of several Special Issues on International Journals and he has been project responsible in several EU research projects dealing with computer vision and image processing.