Hindawi Publishing Corporation
Advances in Multimedia
Volume 2009, Article ID 640386, 21 pages
doi:10.1155/2009/640386

Review Article

A Survey of Visual Sensor Networks

Stanislava Soro and Wendi Heinzelman

Department of Electrical and Computer Engineering, University of Rochester, Rochester NY 14627, USA

Correspondence should be addressed to Stanislava Soro, soro@ece.rochester.edu

Received 20 March 2009; Accepted 13 May 2009

Recommended by Shiwen Mao

Visual sensor networks have emerged as an important class of sensor-based distributed intelligent systems, with unique performance, complexity, and quality of service challenges. Consisting of a large number of low-power camera nodes, visual sensor networks support a great number of novel vision-based applications. The camera nodes provide information from a monitored site, performing distributed and collaborative processing of their collected data. Using multiple cameras in the network provides different views of the scene, which enhances the reliability of the captured events. However, the large amount of image data produced by the cameras combined with the network's resource constraints require exploring new means for data processing, communication, and sensor management. Meeting these challenges of visual sensor networks requires interdisciplinary approaches, utilizing vision processing, communications and networking, and embedded processing. In this paper, we provide an overview of the current state-of-the-art in the field of visual sensor networks, by exploring several relevant research directions. Our goal is to provide a better understanding of current research problems in the different research fields of visual sensor networks, and to show how these different research fields should interact to solve the many challenges of visual sensor networks.

1. Introduction

Camera-based networks have been used for security monitoring and surveillance for a very long time. In these networks, surveillance cameras act as independent peers that continuously send video streams to a central processing server, where the video is analyzed by a human operator.

With the advances in image sensor technology, low-power image sensors have appeared in a number of products, such as cell phones, toys, computers, and robots. Furthermore, recent developments in sensor networking and distributed processing have encouraged the use of image sensors in these networks, which has resulted in a new ubiquitous paradigm—visual sensor networks. Visual sensor networks (VSNs) consist of tiny visual sensor nodes called camera nodes, which integrate the image sensor, embedded processor, and wireless transceiver. Following the trends in low-power processing, wireless networking, and distributed sensing, visual sensor networks have developed as a new technology with a number of potential applications, ranging from security to monitoring to telepresence.

In a visual sensor network a large number of camera nodes form a distributed system, where the camera nodes are able to process image data locally and to extract relevant information, to collaborate with other cameras on the application-specific task, and to provide the system's user with information-rich descriptions of captured events. With current trends moving toward development of distributed processing systems and with an increasing number of devices with built-in image sensors, a question of how these devices can be used together appears [1]. There are several specific questions that have intrigued the research community. How can the knowledge gained from wireless sensor networks be used in the development of visual sensor networks? What kind of data processing algorithms can be supported by these networks? What is the best way to manage a large number of cameras in an efficient and scalable manner? What are the most efficient camera node architectures? Inspired by the tremendous potential of visual sensor networks as well as by the current progress in this research field, we provide in this paper an overview of the current research directions, challenges, and potential applications for visual sensor networks.

Several survey papers on multimedia sensor networks and visual processing can be found in the current literature. In [2], Misra et al. provide a survey of proposed solutions for different layers of the network protocol stack used for multimedia transmission over the wireless medium. Charfi et al. [3] provide a survey on several challenging issues in the design of visual sensor networks design, including coverage requirements, network architectures, and energy-aware data communication and processing. Here, we go one step further, by discussing these and other aspects of visual sensor networks in more detail and taking a multidisciplinary look at these topics. An extensive survey of wireless multimedia sensor networks is provided in [4], where Akyildiz et al. discuss various open research problems in this research area, including networking architectures, single layer and cross-layer communication protocol stack design, and multimedia sensor hardware. Here, we discuss similar problems, but considering visual sensor networks as distributed systems of embedded devices, highly constrained in terms of available energy, bandwidth resources and with limiting processing capabilities. Thus, we are focusing on the low power and low complexity aspects of visual sensor networks. Considering that many aspects of visual sensor networks, such as those related to the design of the networking protocol stack or data encoding techniques in the application layer have already been thoroughly discussed in [2, 4], we focus here on other aspects of data communication, by emphasizing the need for collaborative data communication and sensor management in visual sensor networks. Thus, this paper complements these other survey papers and can be a valuable source of information regarding the state-of-the-art in several research directions that are vital to the success of visual sensor networks.

2. Characteristics of Visual Sensor Networks

One of the main differences between visual sensor networks and other types of sensor networks lies in the nature of how the image sensors perceive information from the environment. Most sensors provide measurements as 1D data signals. However, image sensors are composed of a large number of photosensitive cells. One measurement of the image sensor provides a 2D set of data points, which we see as an image. The additional dimensionality of the data set results in richer information content as well as in a higher complexity of data processing and analysis.

In addition, a camera's sensing model is inherently different from the sensing model of any other type of sensor. Typically, a sensor collects data from its vicinity, as determined by its sensing range. Cameras, on the other hand, are characterized by a directional sensing model—cameras capture images of distant objects/scenes from a certain direction. The 2D sensing range of traditional sensor nodes is, in the case of cameras, replaced by a 3D viewing volume (called field of view, or FoV).

Visual sensor networks are in many ways unique and more challenging compared to other types of wireless sensor networks. These unique characteristics of visual sensor networks are described next.

2.1. Resource Requirements. The lifetime of battery-operated camera nodes is limited by their energy consumption, which is proportional to the energy required for sensing, processing, and transmitting the data. Given the large amount of data generated by the camera nodes, both processing and transmitting image data are quite costly in terms of energy, much more so than for other types of sensor networks. Furthermore, visual sensor networks require large bandwidth for transmitting image data. Thus both energy and bandwidth are even more constrained than in other types of wireless sensor networks.

2.2. Local Processing. Local (on-board) processing of the image data reduces the total amount of data that needs to be communicated through the network. Local processing can involve simple image processing algorithms (such as background substraction for motion/object detection, and edge detection) as well as more complex image/vision processing algorithms (such as feature extraction, object classification, scene reasoning). Thus, depending on the application, the camera nodes may provide different levels of intelligence, as determined by the complexity of the processing algorithms they use [5]. For example, low-level processing algorithms (such as frame differencing for motion detection or edge detection algorithms) can provide a camera node with the basic information about the environment, and help it decide whether it is necessary to transmit the captured image or whether it should continue processing the image at a higher level. More complex vision algorithms (such as object feature extraction, object classification, etc.) enable cameras to reason about the captured phenomena, such as to provide basic classification of the captured object. Furthermore, the cameras can collaborate by exchanging the detected object features, enabling further processing to collectively reason about the object's appearance or behavior. At this point the visual sensor network becomes a user-independent, intelligent system of distributed cameras that provides only relevant information about the monitored phenomena. Therefore, the increased complexity of vision processing algorithms results in highly intelligent camera systems that are oftentimes called smart camera networks [6].

In order to extract necessary information from different images, a camera node must employ different image processing algorithms. One specific image processing algorithm cannot achieve the same performance for different types of images—for example, an algorithm for face extraction significantly differs from algorithm for vehicle detection. However, oftentimes it is impossible to keep all the necessary image processing algorithms in the constrained memory of a camera node. One solution to this problem is to use mobile agents—a specific piece of software dispatched by the sink node to the region of interest [7]. Mobile agents collect and aggregate the data using a specific image algorithm and send the processed data back to the sink. Furthermore, the mobile

agents can migrate between the nodes in order to perform the specific task, thereby performing distributed information processing [8]. In this way, the amount of data sent by the node, as well as the number of data flows in the network, can be significantly reduced.

2.3. Real-Time Performance. Most applications of visual sensor networks require real-time data from the camera nodes, which imposes strict boundaries on maximum allowable delays of data from the sources (cameras) to the user (sink). The real-time performance of a visual sensor network is affected by the time required for image data processing and for the transmission of the processed data throughout the network. Constrained by limited energy resources and by the processing speed of embedded processors, most camera nodes have processors that support only lightweight processing algorithms. On the network side, the real-time performance of a visual sensor network is constrained by the wireless channel limitations (available bandwidth, modulation, data rate), employed wireless standard, and by the current network condition. For example, upon detection of an event, the camera nodes can suddenly inject large amounts of data in the network, which can cause data congestion and increase data delays.

Different error protection schemes can affect the real-time transmission of image data through the network as well. Commonly used error protection schemes, such as automated-repeat-request (ARQ) and forward-error-correction (FEC) have been investigated in order to increase the reliability of wireless data transmissions [9]. However, due to the tight delay constraints, methods such as ARQ are not suitable to be used in visual sensor networks. On the other hand, FEC schemes usually require long blocks in order to perform well, which again can jeopardize delay constraints.

Finally, multihop routing is the preferred routing method in wireless sensor networks due to its energy-efficiency. However, multihop routing may result in increased delays, due to queueing and data processing at the intermediate nodes. Thus, the total delay from the data source (camera node) to the sink increases with the number of hops on the routing path. Additionally, bandwidth becomes a scarce resource in multihop networks consisting of traditional wireless sensor nodes. In order to support the transmission of real-time data, different wireless modules that provide larger bandwidths (such as those based on IEEE 802.11 b,g,n) can be considered.

2.4. Precise Location and Orientation Information. In visual sensor networks, most of the image processing algorithms require information about the locations of the camera nodes as well as information about the cameras' orientations. This information can be obtained through a camera calibration process, which retrieves information on the cameras' intrinsic and extrinsic parameters (explained in the Section 5.1). Estimation of calibration parameters usually requires knowledge of a set of feature point correspondences among the images of the cameras. When this is not provided, the cameras can be calibrated up to a similarity transformation [10], meaning that only relative coordinates and orientations of the cameras with respect to each other can be determined.

2.5. Time Synchronization. The information content of an image may become meaningless without proper information about the time at which this image was captured. Many processing tasks that involve multiple cameras (such as object localization) depend on highly synchronized cameras' snapshots. Time synchronization protocols developed for wireless sensor networks [11] can be successfully used for synchronization of visual sensor networks as well.

2.6. Data Storage. The cameras generate large amounts of data over time, which in some cases should be stored for later analysis. An example is monitoring of remote areas by a group of camera nodes, where the frequent transmission of captured image data to a remote sink would quickly exhaust the cameras' energy resources. Thus, in these cases the camera nodes should be equipped with memories of larger capacity in order to store the data. To minimize the amount of data that requires storage, the camera node should classify the data according to its importance by using spatiotemporal analysis of image frames, and decide which data should have priority to be stored. For example, if an application is interested in information about some particular object, then the background can be highly compressed and stored, or even completely discarded [12].

The stored image data usually becomes less important over time, so it can be substituted with newly acquired data. In addition, reducing the redundancy in the data collected by cameras with overlapped views can be achieved via local communication and processing. This enables the cameras to reduce their needs for storage space by keeping only data of unique image regions. Finally, by increasing the available memory, more complex processing tasks can be supported on-board, which in return can reduce data transmissions and reduce the space needed for storing processed data.

2.7. Autonomous Camera Collaboration. Visual sensor networks are envisioned as distributed and autonomous systems, where cameras collaborate and, based on exchanged information, reason autonomously about the captured event and decide how to proceed. Through collaboration, the cameras relate the events captured in the images, and they enhance their understanding of the environment. Similar to wireless sensor networks, visual sensor networks should be data-centric, where captured events are described by their names and attributes. Communication between cameras should be based on some uniform ontology for the description of the event and interpretation of the scene dynamics [13].

Table 1: Applications of visual sensor networks.

General application	Specific application
Surveillance	Public places
	Traffic
	Parking lots
	Remote areas
Environmental monitoring	Hazardous areas
	Animal habitats
	Building monitoring
Smart homes	Elderly care
	Kindergarten
Smart meeting rooms	Teleconferencing
	Virtual studios
Virtual reality	Telepresence systems
	Telereality systems

3. Applications of Visual Sensor Networks

With the rapid development of visual sensor networks, numerous applications for these networks have been envisioned, as illustrated in the Table 1. Here, we mention some of these applications.

(i) Surveillance: Surveillance has been the primary application of camera-based networks for a long time, where the monitoring of large public areas (such as airports, subways, etc.) is performed by hundreds or even thousands of security cameras. Since cameras usually provide raw video streams, acquiring important information from collected image data requires a huge amount of processing and human resources, making it time-consuming and prone to error. Current efforts in visual sensor networking are concentrated toward advancing the existing surveillance technology by utilizing intelligent methods for extracting information from image data locally on the camera node, thereby reducing the amount of data traffic. At the same time, visual sensor networks integrate resource-aware camera management policies and wireless networking aspects with surveillance-specific tasks. Thus, visual sensor networks can be seen as a next generation of surveillance systems that are not limited by the absence of infrastructure, nor do they require large processing resources at one central server. These networks are adaptable to the environment dynamics, autonomous, and able to respond timely to a user's requests by providing an immediate view from any desired viewpoint or by analyzing and providing information from specific, user determined areas.

(ii) Environmental monitoring: Visual sensor networks can be used for monitoring remote and inaccessible areas over a long period of time. In these applications, energy-efficient operations are particularly important in order to prolong monitoring over an extended period of time. Oftentimes the cameras are combined with other types of sensors into a heterogeneous network, such that the cameras are triggered only when an event is detected by other sensors used in the network [14].

(iii) Smart homes: There are situations (such as patients in hospitals or people with disabilities), where a person must be under the constant care of others. Visual sensor networks can provide continuous monitoring of people, and using smart algorithms the network can provide information about the person needing care, such as information about any unusual behavior or an emergency situation.

(iv) Smart meeting rooms: Remote participants in a meeting can enjoy a dynamic visual experience using visual and audio sensor network technology.

(v) Telepresence systems: Telepresence systems enable a remote user to "visit" some location that is monitored by a collection of cameras. For example, museums, galleries or exhibition rooms can be covered by a network of camera nodes that provide live video streams to a user who wishes to access the place remotely (e.g., over the Internet). The system is able to provide the user with any current view from any viewing point, and thus it provides the sense of being physically present at a remote location through interaction with the system's interface [15]. Telereality aims to synthesize realistic novel views from images acquired from multiple cameras [16].

4. Research Directions in Visual Sensor Networks

Visual sensor networks are based on several diverse research fields, including image/vision processing, communication and networking, and distributed and embedded system processing. Thus, the design complexity involves finding the best tradeoff between performance and different aspects of these networks. According to Hengstler and Aghajan [17] the design of a camera-based network involves mapping application requirements to a set of network operation parameters that are generally related to several diverse research fields, including network topology, sensing, processing, communication, and resource utilization.

Due to its interdisciplinary nature, the research directions in visual sensor networks are numerous and diverse. In the following sections we present an overview of the ongoing research in several areas vital to visual sensor networks: vision processing, wireless networking, camera node hardware architectures, sensor management, and middleware, as illustrated in Figure 1. The survey begins by addressing problems in vision processing related to camera calibration. Then, research related to object detection, tracking, and high-level vision processing is discussed. The survey next provides an overview of different networking problems, such as those related to real-time data communication, camera collaboration and route selection. Next, various sensor management policies, which aim to provide balance between vision and networking tasks, are discussed. Since both vision processing and communication tasks are limited by the camera node hardware, an overview of the latest camera node's prototype solutions are provided, along with a description of network architectures for several visual

sensor network testbeds. Finally, an overview of visual sensor networks middleware that bridges the gap between the application and the low level network structure is provided. In the last part of this paper, we provide an overview of some of the many open research problems that lie in the intersections of these different research areas.

5. Signal Processing Algorithms

5.1. Camera Calibration. Obtaining precise information about the cameras' locations and orientations is crucial for many vision processing algorithms in visual sensor networks. The information on a camera's location and orientation is obtained through the calibration process, where this information (presented as the camera's orientation matrix R and translation vector T) is found from the set of feature points that the camera sees.

Calibration of cameras can be done at one processing center, which collects image feature points from all cameras in the system and, based on that, it estimates the calibration parameters for the entire system. However, such a calibration method is expensive in terms of energy and is not scalable, and thus it is not suitable for energy-constrained visual sensor networks. Therefore, visual sensor networks require distributed energy-efficient algorithms for multicamera calibration.

The localization algorithms developed for wireless sensor networks cannot be used for calibration of the cameras since they do not provide sufficient precision, nor do they provide information on the cameras' orientations. The ad hoc deployment of camera nodes and the absence of human support after deployment imposes the need for autonomous camera calibration algorithms. Since usually there is no prior information about the network's vision graph (a graph that provides information about overlapped cameras' FoVs), communication graph, or about the environment, finding correspondences across cameras (presented as a set of points in one camera's image plane that correspond to the points in another camera's image) is challenging and error prone. Ideally, cameras should have the ability to self-calibrate based on their observations from the environment. The first step in this process involves finding sets of cameras that image the same scene points. Finding correspondences among these cameras may require excessive, energy expensive inter-camera communication. Thus, the calibration process of distributed cameras is additionally constrained by the limited energy resources of the camera nodes. Additionally, the finite transmission ranges of the camera nodes can limit communication between them.

Therefore, camera calibration in a visual sensor network is challenged by finding the cameras' precise extrinsic parameters based on existing calibration procedures taken from computer vision, but considering the communication constraints and energy limitations of camera nodes. These calibration methods should cope successfully with changes in the communication graph (caused by variable channel conditions) and changes in the visual graph (due to the loss of cameras or a change in the cameras' positions and orientations).

Calibration based on a known object is a common calibration method from computer vision, that is, widely adopted in visual sensor networks [18, 19]. In [18] Barton-Sweeney et al. present a light-wight protocol for camera calibration based on such an approach, where the network contains a fraction of wireless nodes equipped with CMOS camera modules, while the rest of the nodes use unique modulated LED emissions in order to uniquely identify themselves to the cameras. This calibration method requires distance information among the cameras, which is obtained through finding epipoles (illustrated in Figure 2) among the pairs of cameras. The authors distinguish two cases for estimation of the distances between two cameras, the case when cameras, in addition of observing the common target (node), can see each other, and the case when they cannot see each other. In the first case the distances between the cameras and the node can be determined up to a scale factor [20]. In the second case, the epipoles estimation is based on estimation of fundamental matrix (based on a minimum of 8 points in the common view), which results in noisy data.

Thus, in [18] the authors do not provide fully automatic camera calibration methods, but instead they point out the difficulty of finding appropriate network configurations that can ease the calibration process.

Funiak et al. [19] provide a distributed method for camera calibration based on collaborative tracking of a moving target by multiple cameras. Here, the simultaneous localization and tracking (SLAT) problem is analyzed, which refers to estimation of both the trajectory of the object and the poses of the cameras. The proposed solution to the SLAT problem is based on an approximation of a Kalman filter. The restrictions imposed by the communication network are not considered in the proposed method.

Devarajan et al. [21] add the underlying communication network model into their proposed camera calibration algorithm, thereby analyzing its performances with respect to the calibration accuracy as well as communication overhead. Their calibration procedure is based on the bundle adjustment method that minimizes a nonlinear cost of the camera parameters and a collection of unknown 3D scene points projected on matched image correspondences. The distributed calibration is performed by clusters of cameras that share the same scene points. The simulation results prove the advantage of using distributed over centralized calibration. The average error in the estimated parameters is similar in both cases, but the distributed calibration method requires less time since it performs optimization over a smaller number of estimating parameters. Additionally, the communication burden is smaller and more evenly distributed across the camera nodes in the case of distributed calibration compared to the centralized approach. However, this method includes finding accurate multiimage correspondences, requiring excessive resources and computational burden, which makes this calibration protocol less attractive for resource constrained visual sensor networks.

Most of the algorithms for camera calibration in visual sensor networks are based on existing calibration methods established in computer vision, and rarely are they influenced by the underlying network. Thus, future camera calibration

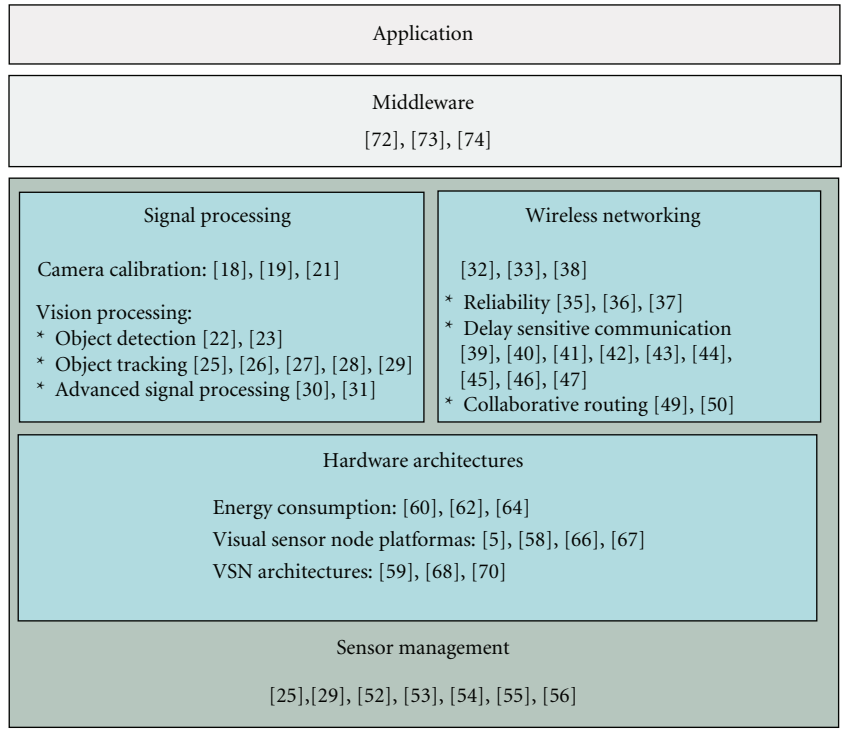


Figure 1: Several research areas that contribute to the development of visual sensor networks.

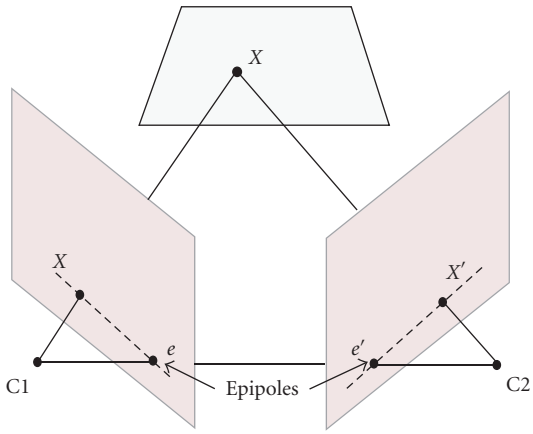


Figure 2: Epipoles of a pair of cameras—the points where the line that connects the centers of the cameras intersects the cameras' image planes [10, 18].

algorithms should explore how the outcome of these calibration algorithms can be affected by the communication constraints and network topology. In particular, it is necessary to find out how multicamera calibration methods can be affected by the underlying networking requirements for reliable and energy efficient intercamera communication. Such an analysis would provide an insight into the trade-offs between the desired calibration precision and cost for achieving it.

Also, the calibration methods should be robust to the network's dynamics; for example, considering how the addition of new cameras or the loss of existing cameras affect the calibration process. Above all, the calibration algorithms should be light-weight, meaning that they should not be based on extensive processing operations. Instead, they should be easily implementable on the hardware platforms of existing camera nodes. Due to the ad hoc nature of visual sensor networks, future research is required to develop camera calibration algorithms that determine precise calibration parameters using a fully automatic approach that requires minimal or no a priori knowledge of network distances, network geometry or corresponding feature points.

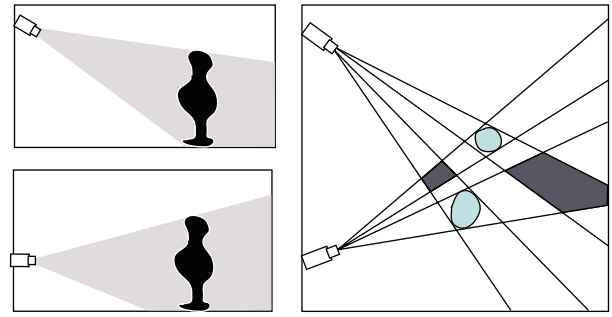
5.2. Vision-Based Signal Processing. The appearance of small CMOS image sensors and the development of distributed wireless sensor networks opens the door to a new era in embedded vision processing. The challenge is how to adapt existing vision processing algorithms to be used in resource-constrained distributed networks of mostly low-resolution cameras. The main constraint comes from the amount of data that can be transmitted through the network. Additionally, most vision processing algorithms are developed without regard to any processing limitations. Furthermore, timing constraints of existing algorithms need to be carefully reconsidered, as the data may travel over multiple hops. Finally, many vision processing algorithms are developed for single camera systems, so these algorithms now need to be adapted for multicamera distributed systems.

The limited processing capabilities of camera nodes dictate a need for light-weight vision processing algorithms in visual sensor networks. However, distributed processing of image data and data fusion from multiple image sources requires more intelligent embedded vision algorithms. As the processing algorithms start to become more demanding (such as those that rely on extraction of feature points and feature matching across multiple cameras' views) the processing capabilities can become a bottleneck. Considering the hierarchical model for vision processing provided in [17], here we describe the main vision processing tasks for visual sensor networks.

5.2.1. Object Detection and Occupancy Reasoning. The initial phase of visual data processing usually involves object detection. Object detection may trigger a camera's processing activity and data communication. Object detection is mostly based on light-weight background substraction algorithms and presents the first step toward collective reasoning by the camera nodes about the objects that occupy the monitored space.

Many applications of visual sensor networks require reasoning about the presence of objects in the scene. In occupancy reasoning, the visual sensor network is not interested in extracting an individual object's features, but instead extracting the state of the scene (such as information about the presence and quantity of objects in the monitored scene) based on light-weight image processing algorithms. An example of such occupancy reasoning in visual sensor networks is the estimation of the number of people in a crowded scene, as discussed in [22]. Here the estimates are obtained using a planar projection of the scene's visual hull, as illustrated in Figure 3. Since the objects may be occluded, the exact number of objects cannot be determined, but instead lower and upper bounds on the number of objects in each polygon are tracked. The estimated bounds on the number of objects are updated over time using a history tree, so that the lower and upper bounds converge toward the exact number of objects in each polygon.

Determining good camera-network deployments and the number of camera nodes to use is also addressed in recent work on occupancy estimation problems. For example, in [23] Yang et al. study a model for managing (tasking) a set of cameras that collectively reason about the occupancy of the monitored area. Their goal is to provide an upper bound on the number of cameras needed to reason about the occupancy for a given accuracy. This task is performed by minimizing the area potentially occupied by the moving objects. Using the Monte Carlo method, the authors in [23] find the number of cameras necessary to provide a visual hull area for one object. However, in the case of multiple objects in the scene, the visual hull area does not converge to the actual area covered by the objects, due to occlusions. Thus, the authors compare several heuristic approaches (uniform, greedy, clustering, and optimal) for finding a subset of the cameras that minimize the visual hull area for the scenario with multiple objects in the scene.



(a) Two cameras observe a person from different positions. The cameras' cones are swept around the person's silhouette

(b) Polygons obtained as the intersection of planar projections of cones in the case of two objects. Visual hull presents the largest volume in which an object can reside. The dark-colored polygons do not contain any objects

Figure 3: Finding the polygons that contain people based on a projection of the person' silhouettes on the planar scene [22].

Since detection of objects on the scene is usually the first step in image analysis, it is important to minimize the chances of objects' fault detection. Thus, reliability and light-weight operations will continue to be the main concerns of image processing algorithms for object detection and occupancy reasoning.

5.2.2. Object Tracking. Object tracking is a common task for many applications of visual sensor networks. Object tracking is a challenging task since it is computationally intensive and it requires real-time data processing. The basic methods for target tracking include temporal differencing and template correlation matching [24]. Temporal differencing requires finding the regions in frames separated in time that have been changed, and thus it fails if the object stops moving or if it gets occluded. On the other hand, template correlation matching aims to find the region of an image that best correlates to an image template. This method is not robust to changes in the object's appearance, such as object size, orientation, or even light conditions. Sophisticated tracking algorithms, which rely on motion parameter estimation and probability estimates (such as tracking algorithms based on Kalman filtering [25] or particle filtering [26]) are suitable for smart camera networks with advanced processing capabilities.

The availability of multiple views in visual sensor networks improves tracking reliability, but with the price of an increased communication overhead among the cameras. Therefore, in resource-constrained visual sensor networks it is important to use lightweight processing algorithms and to minimize the data load that has to be communicated among the cameras. Lau et al. [27] provide an example of a simple algorithm for tracking multiple targets based on hue histograms. After background substraction and segmentation, the histogram of detected blobs in the scene is found and then compared with the histograms found for previous frames in order to track the objects.

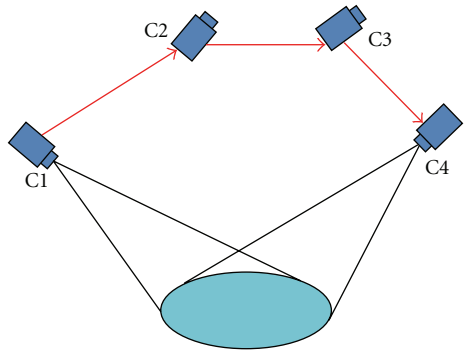


Figure 4: Cameras C1 and C4 observe the same part of the scene, but are not in communication range of each other. Thus, data routing is performed over other camera nodes [30].

Ko and Berry [28] investigate a distributed scheme for target tracking in a multicamera environment. Their collaborative strategy is based on establishing information links between the cameras that detect the target (initiators) and their neighboring cameras that can share information about the tracked target. The cameras extract several target features (edge histogram, UV color histogram, and local position of target) which are correlated across the nodes in order to decide whether information links should be established between the nodes. Such an approach improves the accuracy of the target detection and significantly reduces the communication load.

The success of the proposed tracking algorithms can be jeopardized in the case when the tracked objects are occluded. Object occlusion, which happens when a camera looses sight of an object due to obstruction by another object, is an unavoidable problem in visual sensor networks. Although in most cases the positions of moving occluders cannot be predicted, still it is expected that a multicamera system can handle the occlusion problem more easily due to providing multiple object views. This problem is discussed in [29], where the authors examine the dependance of single object tracking on prior information about the movement of the tracked object and about static occluders. The real challenge in visual sensor networks however, is to avoid losing the tracked object due to occlusions in the situation when not all cameras are available for tracking at the same time. Thus, future research should be directed toward examining the best sensor management policies for selecting camera nodes that will enable multiple target views, thereby reducing the chances of occlusion while using the minimum number of cameras.

5.2.3. Advanced Signal Processing in VSNs. Many novel applications of visual sensor networks are based on advanced vision processing that provides a thorough analysis of the objects' appearances and behaviors, thereby providing a better understanding of the relationships among the objects and situation awareness to the user. In these applications the objective is to provide the automated image understanding by developing efficient computational methods based on principled fundamental issues in automated image understanding. These issues include providing and understanding the performance of methods for object recognition, classification, activity recognition, context understanding, background modeling, and scene analysis.

In such an application a visual sensor network can be used to track human movements but also to interpret these movements in order to recognize semantically meaningful gestures. Human gesture analysis and behavior recognition have gained increasing interest in the research community and are used in a number of applications such as surveillance, video conferencing, smart homes, and assisted living. Behavior analysis applications require collaboration among the cameras, which exchange preprocessed, high level descriptions of the observed scene, rather than the raw image information. In order to reduce the amount of information exchanged between the cameras, research is directed toward finding an effective way of describing the scene and providing the semantic meaning of the extracted data (features). An example of such research is provided in [45], where Teixeira et al. describe a camera-based network that uses symbolic information in order to summarize the motion activity of people. The extracted basic functions of human activity are analyzed using a sensing grammar, which provides the probability likelihood of each outcome. The sequences of basic features of human activity are fed into a inference model, that is, used to reason about the macroscopic behaviors of people—the behavior in some area over a long period of time.

Human behavior interpretation and gesture analysis often use explicit shape models that provide a priori knowledge of the human body in 3D. Oftentimes, these models assume a certain type of body movement, which eases the gesture interpretation problem in the case of body self-occlusion. Recent work of Aghajan and Wu [46] provides a framework for human behavior interpretation based on a 3D human model for estimation of a user's posture from multiple cameras' views. This model is reconstructed from previous model instances and current multiple camera views, and it contains information on geometric body configuration, color/texture of body parts, and motion information. After fitting ellipses to corresponding body parts (segments), human posture is estimated by minimizing the distance between the posture and the ellipses.

Another approach in designing context-aware visual based networks involves using multimodal information for the analysis and interpretation of the objects' dynamics. In addition to low-power camera nodes, such systems may contain other types of sensors such as audio, vibration, thermal, and PIR. By fusing multimodal information from various nodes, such a network can provide better models for understanding an object's behavior and group interactions.

The aforementioned vision processing tasks require extracting features about an event, which in the case of energy and memory constrained camera nodes can be hard or even impossible to achieve, especially in real-time. Thus, although it is desirable to have high-resolution data features, costly feature extractions actually should be limited. This implies the need for finding optimal ways to determine when

Table 2: Representatives of networking protocols used in visual sensor networks.

Criteria	Protocol	Strategy
Reliability		Combined redundant data transmission over multipath routes and error correction algorithms
	Wu and Abouzeid [31]	Multipath cluster based data transmissions combined with error correction at each cluster head
	Chen et al. [32]	Multipath geographical routing and error correction along the routing paths
	Maimour et al. [33]	Comparison of different strategies for load repartition over the multiple routing paths
Delay		Design of delay sensitive MAC and routing protocols, and cross-layer approaches
MAC protocols	DSMAC—Lin et al. [34]	Adjustable sleeping periods of sensor nodes according to the traffic conditions
	DMAC—Lu et al. [35]	Eliminates the delays caused by sleepy nodes that are unaware of current data transmissions
	Ceken [36]	TDMA-based delay aware MAC protocol that provides more time slots for time critical nodes
Routing protocols	SPEED—He et al. [37]	Transmission delay of a packet depends on the distance to the sink and delivery speed
	MMSPEED—Felemban et al. [38]	Multispeed transmission and the establishment of more than one path to the destination
	Lu and Krishnamachari [39]	Joint routing and delay optimization
Cross-layer approaches	Andreopoulos et al. [40]	Capacity-distortion optimization based on several parameters of routing, MAC, and physical layer
	Van der Schaar and Turaga [41]	Packetization and packet retransmission optimization
	Wang et al. [42]	Cross layer protocol for adaptive image transmission for quality optimization of wavelet transformed image
Collaborative image routing		Using spatiotemporal information form multiple correlated data sources
	Obraczka et al. [43]	Communication overhead reduction by collective reasoning based on correlated data
	Medeiros et al. [44]	Cluster-based object tracking

feature extraction tasks can be performed and when they should be skipped or left to other active cameras, without degrading overall performance. Also, most of the current work still use a centralized approach for data acquisition and fusion. Thus, future research should be directed toward migrating the process of decision making to the sensors, and toward dynamically finding the best camera node that can serve as a fusion center to combine extracted information from all active camera nodes.

6. Communication Protocols

Communication protocols for the "traditional" wireless sensor networks are mostly focused on supporting requirements for energy-efficiency in the low data rate communications. On the other hand, in addition to energy-efficiency, visual sensor networks are constrained with much tighter quality of service (QoS) requirements compared to "traditional" wireless sensor networks. Some of the most important QoS requirements of visual sensor networks, such as requirements for low data delay and data reliability, are not the primary concerns in the design of communication protocols for "traditional" wireless sensor networks. Additionally, the sensing characteristics of image sensors can also affect the design of communication protocols for visual sensor networks. For example, in [30], we found that the performance of a coverage-aware routing protocol that was initially developed for wireless sensor networks can change when such a protocol is applied to a visual sensor network. This change in protocol behavior is caused by the fact that distant out-of-communication-range cameras can still observe (cover) a common part of the scene (illustrated in Figure 4), which can influence how this protocol selects routing paths in the network. Thus, the communication protocols developed for traditional wireless sensor networks cannot be simply reused in visual sensor networks.

An event captured by a visual sensor network can trigger the injection of large amounts of data into the network from multiple sources. Each camera can inject variable amounts of data into the network, depending on the data processing (image processing algorithm, followed by the data compression and error correction). The end-to-end data transmissions should satisfy the delay guarantees, thus

requiring stable data routes. At the same time, the choice of routing paths should be performed such that the available network resources (e.g., energy and channel bandwidth) are efficiently balanced across the network.

Beside the energy efficiency and strict QoS constraints, the used data communication model can be influenced by the required quality of the image data provided by the visual sensor network. For example, in [47], Lecuire et al. use an adaptive energy-conserving data transmission model, where nodes, based on their remaining energies, decide whether they will forward packets of a certain priority. The packet priority is defined either based on the resolution level (subband) of the image's wavelet transformation or based on the magnitude of the wavelet coefficients. In order to avoid situations where the data packets are dropped near the data sink, this transmission scheme decreases the probability of packet discarding as the packet approaches the sink. This transmission scheme offers a trade-off in consumed energy versus reconstructed image quality, and it demonstrates the advantage of the magnitude-based prioritization scheme over the resolution level scheme.

Another important aspect in the design of communication protocols for visual sensor networks includes the support for camera collaboration on a specific task. Therefore, the reliable transmission of delay constrained data obtained through collaboration of a number of camera nodes is the main focus of the networking protocols for visual sensor networks. Thus, we further discuss the influence of requirements for reliability, latency, and collaborative processing to the design of data communication protocols for visual sensor networks. Table 2 provides an overview of the networking protocols that are discussed throughout this section with respect to reliability, latency, and collaborative data routing.

6.1. Reliability. Reliable data transport is one of the main QoS requirements of visual sensor networks. In wireless sensor networks, the transport layer of the traditional protocol stack is not fully developed, since the traditional functions of this layer that should provide reliable data transport, such as congestion control, are not a primary concern in low data, low duty-cycle wireless sensor networks. However, the bursty and bulky data traffic in visual sensor networks imposes the need for establishing mechanisms that provide reliable data communication over the unreliable channels across the network.

The standard networking protocols designed to offer reliable data transport are not suitable for visual sensor networks. The commonly used transport protocol TCP cannot be simply reused in wireless networks, since it cannot distinguish between data losses due to network congestion and due to poor wireless channel conditions. In wireless sensor networks, providing reliability oftentimes assumes data retransmissions, which introduce intolerable delays for visual sensor networks. For example, protocols such as Pump Slowly Fetch Quickly (PSFQ) [48] enable the fast recovery of lost data from the local neighborhood using selective NACKs, however, it assumes that the data is lost only due to the channel conditions, and not due to data congestion, basically assuming transmissions of small data amounts through the network.

Data routing over multiple paths is oftentimes considered as a way to reduce the correlations among the packet losses and to spread the energy consumption more evenly among the cameras. Since data retransmissions increase latency in the network, Wu and Abouzeid [31] propose a transport scheme that combines multipath diversity and Reed-Solomon error correction in order to increase data reliability. In their proposed model, the data source transmits several copies of the same data over multiple paths, which converge to the cluster head. Each cluster head compares the received data copies, and it retransmits the error-corrected version of the data over multiple paths toward the next cluster head. Since the error correction is performed at the cluster heads, this transmission scheme improves the quality of the received image data at the sink (measured by PSNR). Another protocol that aims to increase the reliability of transmitted data over multiple paths is presented by Chen et al. [32]. Here, multiple routing paths are established based on the proposed directional geographical routing (DGR) algorithm that, combined with FEC coding, provides more reliable data transmission compared to single-path routing, and it achieves better performance in overall delay and quality of video data at the sink.

Visual sensor networks can experience significant loses of data due to network congestion. As a way to control data congestion in wireless multimedia networks, Maimour et al. [33] explore several strategies for load repartition on multiple source-sink paths. They compare simple strategies that uniformly distribute the traffic from the data source on all available paths with more complex strategies that use explicit notifications from the congested nodes in order to balance traffic on available paths, while keeping the sending rate unchanged.

Congestion control is a dominant problem in the design of reliable protocols for visual sensor networks. Considering that multimedia data can tolerate a certain degree of loss [49], congestion control mechanisms should provide a trade-off between the quality of the data received from the cameras and the energy expense for transmitting this data. Having concurrent data flows increases the data reliability, but it also greatly increases the transmission cost. Thus, further evaluation is needed to clarify the trade-offs between data reliability and data redundancy in multipath routing schemes for visual sensor networks. Furthermore, most of the described data transmission schemes neglect the requirements for low delays. Thus, we further discuss this QoS requirement of visual sensor networks in the next subsection.

6.2. Delay Sensitive Communication Protocols. Real-time data delivery is a common requirement for many applications of visual sensor networks. Data delays can happen in different layers of the network protocol stack, by unsynchronized interaction between different layers of stack, and delay can be further increased by the wireless channel variability. Thus,

the design of different communication layers of the network protocol stack should be carefully considered in order to improve the data latency in the network.

The rising needs of delay-sensitive applications in wireless sensor networks have caused the appearance of a number of energy-efficient delay-aware MAC protocols. The main idea behind these protocols is to reduce the sleep delays of sensor nodes operating in duty cycles, and to adapt the nodes' duty cycles according the network traffic. Since there is already a comprehensive survey on the design of MAC protocols for multimedia applications in wireless sensor networks [2], we will not cover these protocols in detail, but instead we will mention some of the most representative delay-aware MAC protocols.

The SMAC [50] protocol developed by Ye et al. was among the first MAC protocols that explored adaptive listening in order to reduce multihop latency due to periodic sleep. (In adaptive listening, a node that overhears its neighbors transmission wakes up at the end of that transmission, so that it can receive a message, if it is the next hop for its neighbor transmission.) In the DSMAC [34] protocol, Lin et al. further improve the latency problem of SMAC by allowing the sensor nodes to dynamically change their sleeping intervals in order to adjust to the current traffic conditions. In this way, the latency is reduced in networks with high traffic load, while still supporting the energy efficiency when network traffic is low. In the DMAC [35] protocol, Lu et al. further explore the adaptive listening mechanism, pointing out the data forwarding interruption problem, which happens due to the limited overhearing range of the nodes, so that a node can be out of range for both sender and receiver and thus unaware of the ongoing data transmission. Such nodes go to sleep, which causes the interruption in data forwarding. The DMAC protocol eliminates the sleeping delays by providing the same schedule (receive-transmit-sleep cycles) to the nodes with the same depth in the data gathering tree. These protocols are contention-based, so they provide only best effort service. Other authors favor scheduling-based MAC protocols, as a way to avoid data delays and data loses due to channel contention and packet collisions. One such MAC protocol is presented by Ceken [36], where sensor nodes follow a TDMA schedule for data transmissions, but the delay-critical sensor nodes can request extra time slots from the central node in the case when their queues exceed a certain threshold.

Finding routing strategies that enable data delivery within a certain time delay is an extremely hard problem. He et al. developed the SPEED protocol [37] for real-time communication in multihop wireless sensor networks. Since the end-to-end delay in a multihop network depends on the distance a packet travels, SPEED routes packets according to the packet's maximum delivery speed, defined as the rate at which the packet should travel along a straight line to the destination. Thus, SPEED determines the transmission delay of the packet considering its end-to-end distance and its delivery speed. However, such a routing scheme is not scalable, as the maximum delivery speed cannot guarantee that the packet will arrive before its delay deadline in larger networks. This issue is addressed in [38], where Felemban et al. propose MMSPEED, where nodes can forward packets with a higher (adjustable) speed over the multiple paths if it appears that the packet cannot meet its delay deadline. However, underlying network management policies (discussed in Section 7) that regulate nodes' activities have a large impact on the packets' delivery latency. Thus, the data latency problem in visual sensor networks should be further analyzed considering the nodes' resource-aware scheduling policies.

Such an approach is taken in [39], where Lu and Krishnamachari look into the joint problem of finding the routes and nodes activity schedules that provide the minimum average latency for current active data flows. It is assumed an FDMA channel model, which enables simultaneous packet transmissions from neighboring nodes with minimized interference. The proposed solution finds a number of disjoint paths over the delay graph constructed by considering the finite delays at each node between the reception and retransmission of a packet in preassigned time slots.

The data delays at different layers of the network protocol stack may be caused by various factors (channel contention, packet retransmissions, long packet queues, nodes' failure, and network congestion). The cross-layer approaches that consider close interactions between different layers of the protocol stack enable the design of frameworks that support delay-sensitive applications of visual sensor networks.

Andreopoulos et al. [40] propose a cross-layer optimization algorithm that aims to find several parameters that maximize the network's capacity-distortion utility function, while considering delay-constrained streaming in a wireless network. The proposed end-to-end optimization algorithm chooses the optimum routing path, the maximum number of retransmissions at the MAC layer as well as the best modulation scheme at the physical layer (considering thereby the available channel bandwidth and data rates). The proposed optimization model assumes the existence of predetermined time reservations per link with contention free access to the wireless channel. Van der Schaar and Turaga [41] propose cross-layer optimized packetization and retransmission strategies constrained by delay requirements for video delivery in wireless networks. Similarly to [40, 41] is based on rate-distortion optimization algorithms, and in both works the energy constrained resources of nodes in the network are not considered. Such a cross-layer resource allocation problem is analyzed in [42], where Wang et al. discuss the adaptive image transmission scheme that optimizes image quality over a multihop network while considering multihop path conditions such as delay constraints and the probability of delay violation. The design guideline of this work lies in the fact that the information about the position of coefficients in a wavelet transformed image have higher importance and thus higher protection levels than the information about the coefficients' magnitudes. Optimizing the image quality over the multihop network involves finding the optimal source coding rates, which can be translated into the maximum source traffic rate with QoS delay bound.

Cross-layer optimization of the protocol stack enables visual sensor networks to meet various QoS constraints of visual data transmissions, including data communication

within delay bounds. This cross-layer optimization needs also to include different strategies for intra-camera collaborations, which will lead to a reduction of the total data transmitted in the network. We discuss this problem further in the next subsection.

6.3. Collaborative Image Data Routing. In current communication protocols, the camera nodes compete for the network resources, rather than collaborate in order to effectively exploit the available network resources. Thus, the design of communication protocols for visual sensor networks needs to be fundamentally changed, in order to support exchanges of information regarding camera nodes' information contents, which will help to reduce the communication of redundant data and to distribute resources equally among the camera nodes.

Collaboration-based communication should be established between cameras with overlapped FoVs that, based on the spatial-temporal correlation between their images, collectively reason about the events and thus reduce the amount of data and control overhead messages routed through the network [43]. Such a collaboration-based approach for communication is oftentimes used in object tracking applications, where camera nodes are organized into clusters, as for example shown in [44]. Here, the formation of multiple clusters is triggered by the detection of objects. The cluster head node tracks the object, and the cluster head role is assigned to another cluster member once the object is out of the viewing field of the current cluster head. However, in visual sensor networks collaborative clusters can be formed by cameras that have overlapped FoVs, although they can be distant from each other, which can raise questions about the network connectivity. In wireless sensor networks, two nodes are connected if they are able to exchange RF signals. Zhang and Hou [51] prove that if the communication range is at least twice the sensing range, then the complete coverage of a convex area implies that the nodes are connected. However, relation between connectivity and coverage in visual sensor networks needs further investigation, considering the fact that 3D coverage needs to be satisfied and that the area of a camera's coverage usually does not overlap with the transmission range of the camera node.

Finally, supporting data priority has a large effect on the application QoS of visual sensor networks. Camera nodes that detect an event of interest should be given higher priority for data transmissions. In collaborative data processing, camera nodes should collectively decide on data priorities from cameras that provide the most relevant information regarding the captured event. Therefore, protocols that provide differentiated service to support prioritized data flows are needed and must be investigated.

7. Sensor Management

In redundantly deployed visual sensor networks a subset of cameras can perform continuous monitoring and provide information with a desired quality. This subset of active cameras can be changed over time, which enables balancing of the cameras' energy consumption, while spreading the monitoring task among the cameras. In such a scenario the decision about the camera nodes' activity and the duration of their activity is based on sensor management policies. Sensor management policies define the selection and scheduling (that determines the activity duration) of the camera nodes' activity in such a way that the visual information from selected cameras satisfies the application-specified requirements while the use of camera resources is minimized. Various quality metrics are used in the evaluation of sensor management policies, such as the energy-efficiency of the selection method or the quality of the gathered image data from the selected cameras. In addition, camera management policies are directed by the application; for example, target tracking usually requires selection of cameras that cover only a part of the scene that contains the non-occluded object, while monitoring of large areas requires the selection of cameras with the largest combined FoV.

While energy-efficient organization of camera nodes is oftentimes addressed by camera management policies, the quality of the data produced by the network is the main concern of the application. Table 3 compares several camera management policies considering energy efficiency and bandwidth allocation as two quality metrics for camera selection in two common applications—target tracking and monitoring of large scenes.

Monitoring of large areas (such as parking lots, public areas, large stores, etc.) requires complete coverage of the area at every point in time. Such an application is analyzed in [52], where Dagher et al. provide an optimal strategy for allocating parts of the monitored region to the cameras while maximizing the battery lifetime of the camera nodes. The optimal fractions of regions covered by every camera are found in a centralized way at the base station. The cameras use JPEG2000 to encode the allocated region such that the cost per bit transmission is reduced according to the fraction received from the base station. However, this sensor management policy only considers the coverage of a 2D plane, without occlusions and perspective effects, which makes it harder to use in a real situation.

Oftentimes the quality of a reconstructed view from a set of selected cameras is used as a criterion for the evaluation of camera selection policies. Park et al. [53] use distributed look-up tables to rank the cameras according to how well they image a specific location, and based on this they choose the best candidates that provide images of the desired location. Their selection criterion is based on the fact that the error in the captured image increases as the object gets further away from the center of the viewing frustum. Thus, they divide the frustum of each camera into smaller unit volumes (subfrustums). Then, based on the Euclidian distance of each 3D point to the centers of subfrustums that contain this 3D point, they sort the cameras and find the most favorable camera that contains this point in its field of view. The look-up table entries for each 3D location are propagated through the network in order to build a sorted list of favorable cameras. Thus, camera selection is based exclusively on the quality of the image data provided by

Table 3: Comparison of sensor management policies.

Sensor management policy	QoS criteria		Application		Goal of sensor management metric
	Energy efficiency	Bandwidth allocation	Large scene monitoring	Object tracking	
Dagher et al. [52]	Yes	No	Yes	No	Battery lifetime optimization
Park et al. [53]	No	No	Yes	No	Quality of view for every 3D point
Soro and Heinzelman [54]	Yes	No	Yes	No	Exploring trade-offs between the image quality of reconstructed views and energy efficiency
Zamora and Marculescu [55]	Yes	No	No	Yes	Coordinated-wake up policies for energy conservation
Yang and Nahrstedt [56]	No	Yes	No	Yes	Proposed several sensor selection policies (random, event-based, view-based, priority-based) that consider bandwidth constraints
Pahalawatta et al. [25]	Yes	No	No	Yes	Maximize sum of information utility provided by the active sensors subjected to the average energy that can be used by the network
Ercan et al. [29]	No	No	No	Yes	Object occlusions avoidance

the selected cameras, while the resource constraints are not considered.

A similar problem of finding the best camera candidates is investigated in [54]. In this work, we propose several cost metrics for the selection of a set of camera nodes that provide images used for reconstructing a view from a user-specified view point. Two types of metrics are considered: coverage-aware cost metrics and quality-aware cost metrics. The coverage-aware cost metrics consider the remaining energy of the camera nodes and the coverage of the indoor space, and favor the selection of the cameras with higher remaining energy and more redundant coverage. The quality-aware cost metrics favor the selection of the cameras that provide images from a similar view point as the user's view point. Thus, these camera selection methods provide a trade-off between network lifetime and the quality of the reconstructed images.

In order to reduce the energy consumption of cameras Zamora and Marculescu [55] explore distributed power management of camera nodes based on coordinated node wake-ups. The proposed policy assumes that each camera node is awake for a certain period of time, after which the camera node decides whether it should enter the low-power state based on the timeout statuses of its neighboring nodes. Alternatively, camera nodes can decide whether to enter the low-power state based on voting from other neighboring cameras.

Selection of the best cameras for target tracking has been discussed often [25, 29]. In [25] Pahalawatta et al. present a camera selection method for target tracking applications used in energy-constrained visual sensor networks. The camera nodes are selected by minimizing an information utility function (obtained as the uncertainty of the estimated posterior distribution of a target) subject to energy constraints. However, the information obtained from the selected cameras can be lost in the case of object occlusions. This occlusion problem is further discussed in [29], where Ercan et al. propose a method for camera selection in the case when the tracked object becomes occluded by static or moving occluders. Finding the best camera set for object tracking involves minimizing the MSE of the object position's estimates. Such a greedy heuristic for camera selection shows results close to optimal and outperforms naive heuristics, such as selection of the closest set of cameras to the target, or uniformly spaced cameras. The authors here assume that some information about the scene is known in advance, such as the positions of static occluders, and the object and dynamic occluders' prior probabilities for location estimates.

Although a large volume of data is transmitted in visual sensor networks, none of the aforementioned works consider channel bandwidth utilization. This problem is investigated in [56] where Yang and Nahrstedt provide a bandwidth management framework which, based on different camera selection policies and video content, dynamically coordinates the bandwidth requirements among the selected cameras' flows. The bandwidth estimation is provided at the MAC layer of each camera node, and this information is sent to a centralized bandwidth coordinator that allocates the bandwidth to the selected cameras. The centralized bandwidth allocator guarantees that each camera has the minimum bandwidth required, but the flexibility of distributed bandwidth allocation is lost.

In visual sensor networks, sensor management policies are needed to assure balance between the oftentimes opposite requirements imposed by the wireless networking and vision processing tasks. While reducing energy consumption by limiting data transmissions is the primary challenge of energy-constrained visual sensor networks, the quality of the image data and application QoS improve as the network provides more data. In such an environment, the optimization methods for sensor management developed for wireless sensor networks are oftentimes hard to directly apply to visual sensor networks. Such sensor management policies usually do not consider the event-driven nature of visual sensor networks, nor do they consider the unpredictability of data traffic caused by an event detection.

Thus, more research is needed to further explore sensor management for visual sensor networks. Since sensor management policies depend on the underlying networking policies and vision processing, future research lies in the intersection of finding the best trade-offs between these two aspects of visual sensor networks. Additional work is needed to compare the performance of different camera node scheduling sensor policies, including asynchronous (where every camera follows its own on-off schedule) and synchronous (where cameras are divided into different sets, so that in each moment one set of cameras is active) policies. From an application perspective, it would be interesting to explore sensor management policies for supporting multiple applications utilizing a single visual sensor network.

8. Hardware Architectures for Visual Sensor Networks

A typical wireless sensor node has an 8/16-bit microcontroller, limited memory, and it uses short active periods during which it processes and communicates collected data. Limiting a node's "idle" periods (long periods during which a node listens to the channel) and avoiding power-hungry transmissions of huge amounts of data keep the node's energy consumption sufficiently small, so that it can operate for months or even for years. It is desirable to keep the same low-power features in the design of camera nodes, although in this case more energy will be needed for data capture, processing and transmission. Here, we provide an overview of works that analyze energy consumption in visual sensor networks, as well as an overview of current visual sensor node hardware architectures and testbeds.

8.1. Energy Consumption. The lifetime of a battery-operated camera node is limited by its energy consumption, which is determined by the hardware and working mode of the camera node. In order to collect data about energy consumption and to verify camera node designs, a number of camera node prototypes have been recently built and tested. Energy consumption has been analyzed on camera node prototypes built using a wide range of imagers, starting from very low-power, low-resolution camera nodes [57, 58], to web cameras [59, 60] to advanced, high-resolution cameras.

An estimation of the camera node's lifetime can be done based on its power consumption in different tasks, such as image capture, processing, and transmission. Such an analysis is provided in [60], where Margi et al. present results obtained for the power consumption of a visual sensor network testbed consisting of camera nodes built using a Crossbow Stargate [61] board and a Logitech webcam. Each task has an associated power consumption cost and execution time. Several interesting results are reported in [60]. For example, in their setup the time to acquire and process an image takes 2.5 times longer than the time to transmit the compressed image. The energy cost of analyzing the image (via a foreground detection algorithm) and compression of a portion of the image (when an event is detected) is about the same as compression of the full image. Also, they found that transitioning between states can be expensive in terms of energy and time.

In [62] Jung et al. analyze how different operation modes, such as duty-cycle mode and event-driven mode, affect the lifetime of a camera node. The power consumption specifications of the camera node (which consisted of an iMote2 [63] wireless node coupled with an Omnivision OV7649 camera) consider the power consumption profiles of the main components (CPU, radio, and camera) in different operational modes (sleep, idle, and working). The generic power consumption model provided in [62] can be used for the comparison of different hardware platforms in order to determine the most appropriate hardware solution/working mode for the particular application.

Considering the fact that data transmission is the most expensive operation in terms of energy, Ferrigno et al. [64] aim to find the most suitable compression method that provides the best compromise between energy consumption and the quality of the obtained image. Their analysis is drawn from the results of measurements of the current consumption for each state: standby, sensing, processing, connection, and communication. The authors compare several lossy compression methods, including JPEG, JPEG2000, Set Partitioning in Hierarchical Trees (SPIHT), Subsampling (SS) and Discrete Cosine Transform (DCT). The choice of the most suitable compression technique was between SPIHT, which gives the best compression rate and SS, which requires the smallest execution time, has the simplest implementation and assures the best compromise between the compression rate and processing time.

Analysis of the energy consumption of a camera node when performing different tasks [60] and in different working modes [62] is essential for developing effective resource management policies. Understanding the trade-offs between data processing and data communication in terms of energy cost, as analyzed in [64], helps in choosing the best vision processing techniques that provide data of a certain quality while the lifetime of the camera node is prolonged. Analysis of the energy consumption profile helps the selection of hardware components for the specific application. However, the variety of hardware, processing algorithms and networking protocols used in various testbeds makes the comparison of existing camera nodes difficult. Today, there is no systematic overview and comparison of different visual sensor network testbeds from the energy consumption perspective. Therefore, further research should focus on comparing different camera node architectures and visual sensor network testbeds, in order to explore the energy-performance trade-offs.

8.2. Visual Sensor Node Platforms. Today, CMOS image sensors are commonly used in many devices, such as cell phones and PDAs. We can expect widespread use of image sensors in wireless sensor networks only if such networks still preserve the low power consumption profile. Because of energy and bandwidth constraints, low-resolution image sensors are actually preferable in many applications of visual sensor networks. Table 4 compares several prototypes of

Table 4: Comparison of different visual sensor node architectures.

Camera node architecture	Processing unit	Memory	Image sensor	RF transceiver
MeshEye [5]	Atmel ARM7TDMI Thumb (32-bit RISC), 55 MHz	64 KB SRAM and 256 KB Flash; external MMC/SD Flash	Two kilopixel imagers Agilent Technologies ADNS 3060 30×30 pixels (grayscale) and one ADCM 2700 VGA (color)	Chipcon CC2420 IEEE 802.15.4
Cyclops [58]	Atmel ATmega128L and CPLD—Xilinx XC2C256 CoolRunner	512 KB Flash 64 KB SRAM	ADCM-1700 Agilent Technology	IEEE 802.15.4 compliant (MICA2 Mote [65])
SIMD (Single-instruction-multiple-data)-based architecture [66]	Philips IC3D Xetal (for low-level image processing), 8051 MCU (local host for high level image processing and control)	1792B RAM and 64 KB Flash internal on 8051 MCU; dual port RAM 128 KB (shared memory by both processors)	VGA Image Sensor (one or two)	Aquis Grain Zigbee module based on Chipcon CC2420
CMUCam3 [67]	ARM7TDMI (32-bit) 60 MHz	64 KB RAM and 128 KB Flash on MCU, 1 MB AL4V8M440 FIFO Frame Buffer Flash (MMC)	Omnivision OV6620, 352×288 pixels	IEEE 802.15.4 compliant (Telos mote)

visual sensor nodes with respect to the main hardware components such as processors, memory, image sensor, and RF transceiver.

Compared with processors used for wireless sensor nodes, the processing units used in visual sensor node architectures are usually more powerful, with 32-bit architectures and higher processing speed that enables faster data processing. In some architectures [58, 66] a second processor is used for additional processing and control. Since most procesors have small internal memories, additional external Flash memories are used for frame buffering and permanent data storage. Image sensors also tends to provide small format images (CIF format and smaller). However, some implementations [5, 66] use two image sensors to provide binocular vision. For example, the Mesheye architecture [5] uses two low resolution image sensors (kilopixels) and one high resolution (VGA) image sensor located in between the two low resolution image sensors. With one kilopixel imager the camera node can detect the presence of an object in its FoV. Stereo vision from two kilopixel imagers enables estimation of object position and size, thereby providing the region of interest. Finally, a high resolution image of the region of interest can be obtained using the VGA camera.

It is evident that all camera node prototypes shown in Table 4 use IEEE 802.15.4 RF transceivers, which is commonly used in wireless sensor nodes as well. The Chipcon CC2420 radio supports a maximum of 250Kb/s data rate, although the achievable data rate is often much smaller due to packet overhead and the transient states of the transciever. Since such insufficient data rates can be a bottleneck for vision-based applications, future implementations should consider other radio standards with higher data rates, at the cost of increased energy dissipation. Also, by providing a simpler programming interface, the widespread use of visual sensor nodes can be expected. Such an interface is described in [57] where Hengstler and Aghajan present a framework called Wireless Image Sensor Network Application Platform (WiSNAP) for research and development of applications in wireless visual networks. This Matlab-based application development platform contains APIs that provide a user with interfaces to the image sensor and the wireless node. The WiSNAP framework enables simulations of this visual sensor node platform in different applications.

8.3. VSN Architectures—Testbed Research. Testbed implementations of visual sensor networks are an important final step in evaluating processing algorithms and communication protocols. Several architectures for visual sensor networks can be found in the literature.

Among the first reported video-based sensor network systems is Panoptes [59], which consisted of video sensors built from COTS components and software that supports different functions including image capture, compression, filtering, video buffering, and data streaming. Panoptes supports a priority-based streaming mechanism, where the incoming video data is mapped to a number of priorities defined by the surveillance application. Panoptes provides storage and retrieval of video data from sensors, it handles queries from users, and it controls the streaming of events of interest to the user. However, the system does not have real-time support—a user can only select to see past events already stored in the system. Also, there is no interaction between the cameras.

In [68], Kulkarni et al. present SensEye—a heterogeneous multitier camera sensor network consisting of different nodes and cameras in each tier. The SensEye system is designed for a surveillance application, thus supporting tasks

such as object detection, recognition, and tracking. These tasks are performed across three network tiers. The lowest layer, which supports object detection and localization, is comprised of Mote nodes [69], and low-fidelity CMUCam camera sensors. The second tier contains Stargate nodes [61] equipped with web cameras, which are woken up on demand by the camera nodes from the lower tier to continue the object recognition task. The third tier contains sparsely deployed high-resolution pan-tilt-zoom cameras connected to a PC, which performs the object tracking. The SensEye platform proves that task allocation across tiers achieves a reduction in energy compared with a homogeneous platform, while the latency of the network response is close to the latency achieved by an always-on homogeneous system.

Researchers from Carnegie Melon University present a framework for a distributed network of vision-enabled sensor nodes called FireFly Mosaic [70] (illustrated in Figure 5). The FireFly platform is built from FireFly sensor nodes enhanced with vision capabilities using the CMUCam3 vision processing board [67]. The CMUCam3 sensor supports a set of built-in image processing algorithms, including JPEG compression, frame differencing, color tracking, histogramming, and edge detection. Tight global synchronization throughout the network is supported by using an out-of-band AM carrier current radio transmitter and on-board AM radio receiver.

The communication and collaboration of camera nodes is scheduled using a collision free, energy-efficient TDMA-based link layer protocol called RT-Link [71]. In order to support camera group communication (among the cameras with overlapped FoVs) both the network connectivity graph (that considers the links between nodes within communication range, shown in Figure 6(a)) and the camera network graph (that considers the relationships between the cameras' FoVs, Figure 6(b)) are considered. In this way cameras that share part of the view, but are out of each other's communication range can still communicate via other nodes.

The size of the transmitted images with a given resolution is controlled by the quality parameter provided in the JPEG standard, which is used for image compression. The authors noticed that JPEG processing time does not vary significantly with the image quality level, but it changes with image resolution, mostly due to the large I/O transfer time between the camera and the CPU. The authors also measured the sensitivity of the system's tracking performances with the respect to the time jitter, that is, added to the cameras' image capturing time.

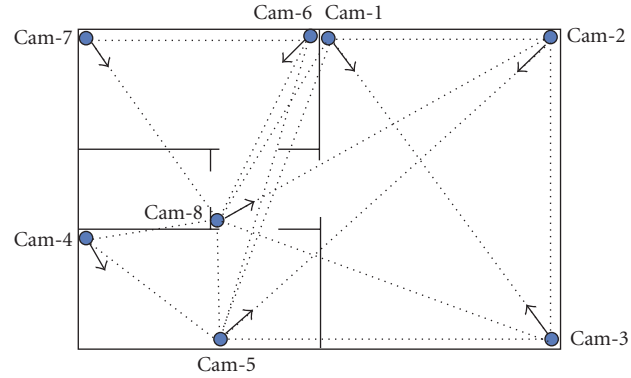
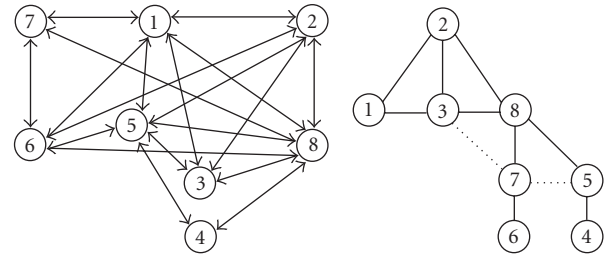


Figure 5: Topology of the visual sensor network, that is, used for testing the FireFly system [70]. The dotted lines represent the communication links between the cameras.



(a) Connectivity graph of the camera nodes from the previous figure. Marked links correspond to the camera network graph

(b) Camera network graph—adjacent links between the cameras indicate that cameras have overlapped FoVs. The dotted lines correspond to the case when the cameras have overlapped views, but cannot communicate directly. The communication schedule must provide message forwarding between these cameras

Figure 6: Connectivity graph and camera network graph of the FireFly system [70].

9. Middleware Support

The increased number of hardware and software platforms for smart camera nodes has created a problem in how to network these heterogeneous devices and how to easily build applications that use these networked devices. The integration of camera nodes into a distributed and collaborative network benefits from a well-defined middleware that abstracts the physical devices into a logical model, providing a set of services defined through standardized APIs that are portable over different platforms. In wireless sensor networks, middleware provides abstractions for the networking and communication services, and the main challenges are associated with providing abstraction support, data fusion and and managing the limited resources [72].

In the case of visual sensor networks, the development of middleware support is additionally challenged by the need for high-level software for supporting complex and distributed vision processing tasks. In [73] this support is provided using agent-oriented middleware, where different image processing tasks are carried out by different agents. The agents are responsible for task execution at the processing unit, they can create new agents, and they can remotely create new agents at other cameras, which is fundamental for distributed organization of a smart camera network.

In [74], Detmold et al. propose using a Blackboard-based middleware approach instead of the popular multiagent approach. In this model, the results of processing of input

video streams are published at the distributed Blackboard component. Thus, the Blackboard acts as a repository of information, where computations are triggered in response to published results. The Blackboard has several interacting levels. The "single scene analysis" provides information derived from object detection and activity analysis (e.g., it produces a "left luggage" hypothesis). The "multi scene analysis" draws conclusions about tracked objects, such as the tracks of people throughout the scene. The "reasoning level" provides higher level hypotheses regarding unusual behavior. Each level contains drivers that process inputs and add them to the level's information space. The information are propagated upwards and shared among the Blackboard levels.

In the future, it is expected that the number of cameras in smart surveillance applications will scale to hundreds or even thousands—in this situation, the middleware will have a crucial role in scaling the network and in integrating the different software components into one automated vision system. In these systems, the middleware should address the system's real-time requirements, together with the other resource (energy and bandwidth) constraints.

10. Open Research Problems in Visual Sensor Networks

The extensive research has been done in the many directions that contribute to the visual sensor networks. However, the real potential of these networks can be reached through a cross-disciplinary research approach that considers all the various aspects of visual sensor networks: vision processing, networking, sensor managemen, and hardware design.

However, in many cases of the existing work there is no coherence between the different aspects of visual sensor networks. For example, networking protocols used in visual sensor networks are mainly adapted from the routing protocols used in traditional wireless sensor networks, and thus do not provide sufficient support for the data-hungry, time-constrained, collaborative communication of visual sensor networks. Similarly, embedded vision processing algorithms used in visual sensor networks are based on existing computer vision algorithms, and thus they rarely consider the constraints imposed by the underlying wireless network.

Thus, future efforts should be directed toward finding ways to minimize the amount of data that has to be communicated, by finding ways to describe captured events with the least amount of data. Additionally, the processing should be lightweight—information rich descriptors of objects/scenes are not an option. Hence, the choice of the "right" feature set, as well as support for real-time communication will play a major role in a successfully operated task.

In order to keep communication between cameras minimal, the cameras need to have the ability to estimate whether the information they provide contributes to the monitoring task. In a postevent detection phase, sensor management policies should decide, based on known information from the cameras and the network status, whether more cameras need to be included in the monitoring. In addition, data exchanged between camera nodes should be aggregated in-network at one of the camera nodes, and the decision about the most suitable data fusion center should be dynamic, considering the best view and the communication/fusion cost. However, considering the oftentimes arbitrary deployment of camera nodes, where the cameras' positions and orientations are not known, the problem is to find the best ways to combine these arbitrary views in order to obtain useful information.

In the current literature distributed source coding (DSC) has been extensively investigated as a way to reduce the amount of transmitted data in wireless sensor networks. In DSC, each data source encodes its data independently, without communicating with the other data sources, while joint data decoding is performed at the base station. This model, where sensor nodes have simple encoders and the complexity is brought to the receiver's end, fits well the needs of visual sensor networks. However, many issues have to be resolved before DSC can be practical for visual sensor networks. For example, it is extremely hard to define the correlation structure between different images, especially when the network topology is unknown or without a network training phase. Also, DSC requires tight synchronization between packets sent from correlated sources. Since DSC should be implemented in the upper layers of the network stack, it affects all the other layers below [75]. Thus, the implementation of DSC will also require careful reconsideration of existing cross-layer designs.

From the communication perspective, novel protocols need to be developed that support bursty and collaborative in-network communication. Supporting time-constrained and reliable communication are problems at the forefront of protocol development for visual sensor networks. In order to support the collaborative processing, it is expected that some cameras acts as a fusion centers by collecting and processing raw data from several cameras. Having several fusion centers can affect the data latency throughout the network as well as the amount of the postfusion data. Thus, further research should explore the trade-offs between the ways to combine (fuse) data from multiple sources and latency introduced by these operations.

Furthermore, in order to preserve network scalability and to cope with time-constrained communication, there is a need for developing time-aware sensor management policies that will favor utilization of those cameras that can send data over multihop shortest delay routes. Such communication should support priority differentiation between different data flows, which can be determined based on vision information and acceptable delays for the particular data.

In the future we can expect to see various applications based on multimedia wireless networks, where camera nodes will be integrated with other types of sensors, such as audio sensors, PIRs, vibration sensors, light sensors, and so forth. By utilizing these very low-cost and low-power sensors, the lifetime of the camera nodes can be significantly prolonged. However, many open problems appears in such multimedia networks. The first issue is network deployment, whereby it is necessary to determine network architecture and the

numbers of different types of sensors that should be used in a particular application, so that all of the sensors are optimally utilized while at the same time the cost of the network is kept low. Such multimedia networks usually employ a hierarchical architecture, where ultra-low power sensors (such as microphones, PIRs, vibration, or light sensors) continuously monitor the environment over long periods of time, while higher-level sensors, such as cameras sleep most of the time. When the lower-level sensors register an event, they notify higher-level sensors about it. Such a hierarchical model (as seen in [68], e.g.) tends to minimize the amount of communication in the network. However, it is important to reduce the number of false and missed alarms at the low-level sensors, so that the network reliability is not jeopardized. Thus, it is important to precisely define an event at the lower-level sensors that cameras can interpret without ambiguity. A high-level node acting as a data collector should be able to perform multimodal fusion of data received from different types of sensors, in order to reason about captured events and decide an appropriate course of action. The reliability of multimodal data fusion thus depends on the accuracy of the data provided by each sensor modality, so the data from different types of sensors can be associated with different weights before the data fusion.

The growing trend of deploying an increasing number of smart sensors in people's everyday lives poses several privacy issues. We have not discussed this problem in this paper, but it is clear that this problem is a source of concern for many people who can benefit from visual sensor networks, as information about their private life can be accessed through the network. The main problem is that the network can take much more information, such as private information, than it really needs in order to perform its tasks. As pointed out in [76], there are several ways to work around this problem. The most radical solution is to exclude cameras from the network, using only nonimaging sensors. However, many situations cannot be resolved without obtaining image data from the area. Thus, the solutions where cameras perform high-level image analysis and provide descriptive information instead of raw images are favorable. The user can be contacted by the system only on occasions when the system is not sure how to react (e.g., if an unknown face is detected in the house). In the future, people will most probably need to sacrifice a bit of their privacy if they want to benefit from smart applications of visual sensor networks. However, privacy and security should be seriously addressed in all future designs of visual sensor networks.

Based on the work reviewed in this paper, we notice that current research trends in visual sensor networks are divided into two directions. The first direction leads toward the development of visual sensor networks where cameras have large processing capabilities, which makes them suitable for use in a number of high-level reasoning applications. Research in this area is directed toward exploring ways to implement existing vision processing algorithm onto embedded processors. Oftentimes, the networking and sensor management aspects are not considered in this approach. The second direction in visual sensor networks research is motivated by the existing research in wireless sensor networks. Thus, it is directed toward exploring the methods that will enable the network to provide small amounts of data from the camera nodes that are constrained by resource limitations, such as remaining energy and available bandwidth. Thus, such visual sensor networks are designed with the idea of having data provided by the network of cameras for long periods of time.

We believe that in the future these two directions will converge toward the same path. Currently, visual sensor networks are limited by their hardware components (COTS) that are not fully optimized for embedded vision processing applications. Future development of faster, low-power processing architectures and ultra low-power image sensors will open a door toward a new generation of visual sensor networks with better processing capabilities and lower energy consumption. However, the main efforts in the current research of visual sensor networks should be directed toward integrating vision processing tasks and networking requirements. Thus, future directions in visual sensor networks research should be aimed at exploring the following interdisciplinary problems.

(i) How should vision processing tasks depend on the underlying network conditions, such as limited bandwidth, limited (and potentially time-varying) connectivity between camera nodes or data loss due to varying channel conditions?

(ii) How should the design of network communication protocols be influenced by the vision tasks? For example, how should different priorities be assigned to data flows to dynamically find the smallest delay route or to find the best fusion center?

(iii) How should camera nodes be managed, considering the limited network resources as well as both the vision processing and networking tasks, in order to achieve application-specific QoS requirements, such as those related to the quality of the collected visual data or coverage of the monitored area?

In the end, widespread use of visual sensor networks depends on the programming complexity of the system, which includes implementation of both vision processing algorithms as well as networking protocols. Therefore, we believe that development of middleware for visual sensor networks will have a major role in making these networks widely accepted in a number of applications. We can envision that in the future visual sensor networks will consist of hundreds or even thousands of camera nodes (as well as other types of sensor nodes) scattered throughout an area. The scalability and integration of various vision and networking tasks for such large networks of cameras should be addressed by future developments of distributed middleware architectures. Middleware should provide an abstraction of underlying vision-processing, networking and shared services (where shared services are those commonly used by both the vision processing and networking tasks and include synchronization service, localization service, or neighborhood discovery service, e.g.). By providing a number of APIs, the middleware will enable easy programming

at the application layer, and the use of different hardware platforms in one visual sensor network.

11. Conclusions

Transmission of multimedia content over wireless and wired networks is a well-established research area. However, the focus of this paper is to survey a new type of wireless networks, visual sensor networks, and to point out the unique characteristics and constraints that differentiate visual sensor networks from other types of multimedia networks. We present an overview of existing work in several research areas that support visual sensor networks. In the coming era of low-power distributed computing, visual sensor networks will continue to challenge the research community because of their complex application requirements and tight resource constraints. We discussed many problems encountered in visual sensor network research caused by the strict resource constraints, including embedded vision processing, data communication, camera management issues, and development of effective visual sensor network testbeds. However, visual sensor networks' potential to provide a comprehensive understanding of the environment and their ability to provide visual information from unaccessible areas will make them indispensable in the coming years.

Many problems still need to be addressed through future research. We discussed some of the open issues not only in the different subfields of visual sensor networks, but, more importantly, in the integration of these areas. Real breakthroughs in visual sensor networks will occur only through a comprehensive solution that considers the vision, networking, management, and hardware issues in concert.

Acknowledgment

This work was supported by the National Science Foundation under Grant #ECS-0428157.

References

[1] P. Bolliger, M. Köhler, and K. Römer, "Facet: towards a smart camera network of mobile phones," in *Proceedings of 1st ACM International Conference on Autonomic Computing and Communication Systems (Autonomics '07)*, 2007.

[2] S. Misra, M. Reisslein, and G. Xue, "A survey of multimedia streaming in wireless sensor networks," *IEEE Communications Surveys and Tutorials*, vol. 10, pp. 18–39, 2008.

[3] Y. Charfi, N. Wakamiya, and M. Murata, "Challenging issues in visual sensor networks," Tech. Rep., Advanced Network Architecture Laboratory, Osaka University, 2007.

[4] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer Networks*, vol. 51, no. 4, pp. 921–960, 2007.

[5] S. Hengstler, D. Prashanth, S. Fong, and H. Aghajan, "MeshEye: a hybrid-resolution smart camera mote for applications in distributed intelligent surveillance," in *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN '07)*, pp. 360–369, 2007.

[6] W. Wolf, B. Ozer, and T. Lv, "Smart cameras as embedded systems," *Computer*, vol. 35, no. 9, pp. 48–53, 2002.

[7] M. Chen, S. Gonzalez, and V. C. M. Leung, "Applications and design issues for mobile agents in wireless sensor networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 20–26, 2007.

[8] M. Chen, T. Kwon, Y. Yuan, Y. Choi, and V. C. M. Leung, "Mobile agent-based directed diffusion in wireless sensor networks," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, 13 pages, 2007.

[9] M. Wu and C. W. Chen, "Multiple bitstream image transmission over wireless sensor networks," in *Proceedings of 2d IEEE International Conference on Sensors*, vol. 2, pp. 727–731, Toronto, Canada, October 2003.

[10] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK, 2000.

[11] K. Römer, P. Blum, and L. Meier, "Time synchronization and calibration in wireless sensor networks," in *Handbook of Sensor Networks: Algorithms and Architectures*, I. Stojmenovic, Ed., pp. 199–237, John Wiley & Sons, New York, NY, USA, 2005.

[12] D. Ganesan, B. Greenstein, D. Perelyubskiy, D. Estrin, and J. Heidemann, "Multi-resolution storage and search in sensor networks," *ACM Transactions on Storage*, vol. 1, pp. 277–315, 2005.

[13] P. Remagnino, A. I. Shihab, and G. A. Jones, "Distributed intelligence for multi-camera visual surveillance," *Pattern Recognition*, vol. 37, no. 4, pp. 675–689, 2004.

[14] T. He, S. Krishnamurthy, L. Luo, et al., "VigilNet: an integrated sensor network system for energy-efficient surveillance," *ACM Transactions on Sensor Networks*, vol. 2, no. 1, pp. 1–38, 2006.

[15] O. Schreer, P. Kauff, and T. Sikora, *3D Video Communication*, John Wiley & Sons, New York, NY, USA, 2005.

[16] N. J. McCurdy and W. Griswold, "A system architecture for ubiquitous video," in *Proceedings of the 3rd Annual International Conference on Mobile Systems, Applications, and Services (Mobisys '05)*, 2005.

[17] S. Hengstler and H. Aghajan, "Application-oriented design of smart camera networks," in *Proceedings of the 1st ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC '07)*, pp. 12–19, 2007.

[18] A. Barton-Sweeney, D. Lymberopoulos, and A. Savvides, "Sensor localization and camera calibration in distributed camera sensor networks," in *Proceedings of the 3rd International Conference on Broadband Communications, Networks and Systems (BROADNETS '06)*, 2006.

[19] S. Funiak, M. Paskin, C. Guestrin, and R. Sukthankar, "Distributed localization of networked cameras," in *Proceedings of the 5th International Conference on Information Processing in Sensor Networks (IPSN '06)*, pp. 34–42, 2006.

[20] C. J. Taylor, "A scheme for calibrating smart camera networks using active lights," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, p. 322, 2004.

[21] D. Devarajan, R. J. Radke, and H. Chung, "Distributed metric calibration of ad hoc camera networks," *ACM Transactions on Sensor Networks*, vol. 2, no. 3, pp. 380–403, 2006.

[22] D. B. Yang, H. H. González-Baños, and L. J. Guibas, "Counting people in crowds with a real-time network of simple image sensors," in *Proceedings of the 9th IEEE International Conference on Computer Vision*, vol. 1, pp. 122–129, Nice, France, October 2003.

[23] D. Yang, J. Shin, A. Ercan, and L. Guibas, "Sensor tasking for occupancy reasoning in a network of cameras," in *Proceedings of 2nd IEEE International Conference on Broadband Communications, Networks and Systems (BaseNets '04)*, 2004.

[24] A. Lipton, H. Fujiyoshi, and R. Patil, "Moving target classification and tracking from real-time video," in *Proceedings of IEEE Image Understanding Workshop*, 1998.

[25] P. V. Pahalawatta, T. N. Pappas, and A. K. Katsaggelos, "Optimal sensor selection for video-based target tracking in a wireless sensor network," in *Proceedings of the International Conference on Image Processing (ICIP '04)*, vol. 2, pp. 3073–3076, 2004.

[26] S. Fleck, F. Busch, and W. Straßer, "Adaptive probabilistic tracking embedded in smart cameras for distributed surveillance in a 3D model," *EURASIP Journal of Embedded Systems*, vol. 2007, Article ID 29858, 17 pages, 2007.

[27] F. Lau, E. Oto, and H. Aghajan, "Color-based multiple agent tracking for wireless image sensor networks," in *Proceedings of the Advanced Concepts for Intelligent Vision Systems (ACIVS '06)*, pp. 299–310, 2006.

[28] T. H. Ko and N. M. Berry, "On scaling distributed low-power wireless image sensors," in *Proceedings of the 39th Annual Hawaii International Conference on System Sciences*, 2006.

[29] A. Ercan, A. E. Gamal, and L. Guibas, "Camera network node selection for target localization in the presence of occlusions," in *Proceedings of the ACM SenSys Workshop on Distributed Smart Cameras*, 2006.

[30] S. Soro and W. B. Heinzelman, "On the coverage problem in video-based wireless sensor networks," in *Proceedings of the 2nd International Conference on Broadband Networks (BROADNETS '05)*, pp. 9–16, 2005.

[31] H. Wu and A. A. Abouzeid, "Error resilient image transport in wireless sensor networks," *Computer Networks*, vol. 50, no. 15, pp. 2873–2887, 2006.

[32] M. Chen, V. C. M. Leung, S. Mao, and Y. Yuan, "Directional geographical routing for real-time video communications in wireless sensor networks," *Computer Communications*, vol. 30, no. 17, pp. 3368–3383, 2007.

[33] M. Maimour, C. Pham, and J. Amelot, "Load repartition for congestion control in multimedia wireless sensor networks with multipath routing," in *Proceedings of the 3rd International Symposium on Wireless Pervasive Computing (ISWPC '08)*, pp. 11–15, 2008.

[34] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '04)*, vol. 3, pp. 1534–1539, 2004.

[35] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks," in *Proceedings of the 18th International Parallel and Distributed Processing Symposium (IPDPS '04)*, pp. 3091–3098, Santa Fe, NM, USA, April 2004.

[36] C. Ceken, "An energy efficient and delay sensitive centralized MAC protocol for wireless sensor networks," *Computer Standards and Interfaces*, vol. 30, no. 1-2, pp. 20–31, 2008.

[37] T. He, J. A. Stankovic, C. Lu, and T. Abdelzaher, "SPEED: a stateless protocol for real-time communication in sensor networks," in *Proceedings of the International Conference on Distributed Computing Systems (ICDCS '03)*, pp. 46–55, 2003.

[38] E. Felemban, C.-G. Lee, and E. Ekici, "MMSPEED: multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 738–753, 2006.

[39] G. Lu and B. Krishnamachari, "Minimum latency joint scheduling and routing in wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 6, pp. 832–843, 2007.

[40] Y. Andreopoulos, N. Mastronarde, and M. van der Schaar, "Cross-layer optimized video streaming over wireless multihop mesh networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 2104–1215, 2006.

[41] M. van der Schaar and D. S. Turaga, "Cross-layer packetization and retransmission strategies for delay-sensitive wireless multimedia transmission," *IEEE Transactions on Multimedia*, vol. 9, no. 1, pp. 185–197, 2007.

[42] W. Wang, D. Peng, H. Wang, and H. Sharif, "Adaptive image transmission with p-v diversity in multihop wireless mesh networks," *International Journal of Electrical, Computer, and Systems Engineering*, vol. 1, no. 1, 2007.

[43] K. Obraczka, R. Manduchi, and J. Garcia-Luna-Aceves, "Managing the information flow in visual sensor networks," in *Proceedings of the 5th International Symposium on Wireless Personal Multimedia Communication*, 2002.

[44] H. Medeiros, J. Park, and A. Kak, "A light-weight event-driven protocol for sensor clustering in wireless camera networks," in *Proceedings of the 1st ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC '07)*, pp. 203–210, 2007.

[45] T. Teixeira, D. Lymberopoulos, E. Culurciello, Y. Aloimonos, and A. Savvides, "A lightweight camera sensor network operating on symbolic information," in *Proceedings of 1st Workshop on Distributed Smart Cameras, Held in Conjunction with ACM SenSys*, 2006.

[46] H. Aghajan and C. Wu, "From distributed vision networks to human behavior interpretation," in *Proceedings of the Behaviour Monitoring and Interpretation Workshop at the 30th German Conference on Artificial Intelligence*, 2007.

[47] V. Lecuire, C. Duran-Faundez, and N. Krommenacker, "Energy-efficient transmission of wavelet-based images in wireless sensor networks," *EURASIP Journal on Image and Video Processing*, vol. 2007, no. 1, 15 pages, 2007.

[48] C.-Y. Wan, A. T. Campbell, and L. Krishnamurthy, "Pump-slowly, fetch-quickly (PSFQ): a reliable transport protocol for sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 862–872, 2005.

[49] M. van der Schaar and P. Chou, *Multimedia over IP and Wireless Networks: Compression, Networking, and Systems*, Academic Press, New York, NY, USA, 2007.

[50] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, vol. 3, pp. 1567–1576, 2002.

[51] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *International Journal of Wireless Ad Hoc and Sensor Networks*, vol. 1, no. 2, pp. 89–124, 2005.

[52] J. C. Dagher, M. W. Marcellin, and M. A. Neifeld, "A method for coordinating the distributed transmission of imagery," *IEEE Transactions on Image Processing*, vol. 15, no. 7, pp. 1705–1717, 2006.

[53] J. Park, P. Bhat, and A. Kak, "A look-up table based approach for solving the camera selection problem in large camera networks," in *Proceedings of the International Workshop on Distributed Smart Cameras (DCS '06)*, 2006.

[54] S. Soro and W. Heinzelman, "Camera selection in visual sensor networks," in *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS '07)*, pp. 81–86, 2007.

[55] N. H. Zamora and R. Marculescu, "Coordinated distributed power management with video sensor networks: analysis, simulation, and prototyping," in *Proceedings of the 1st*

ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC '07), pp. 4–11, 2007.

[56] Z. Yang and K. Nahrstedt, "A bandwidth management framework for wireless camera array," in *Proceedings of the International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV '05)*, pp. 147–152, 2005.

[57] S. Hengstler and H. Aghajan, "WiSNAP: a wireless image sensor network application platform," in *Proceedings of the 2nd International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM '06)*, pp. 7–12, 2006.

[58] M. Rahimi, R. Baer, O. I. Iroezi, et al., "Cyclops: in situ image sensing and interpretation in wireless sensor networks," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, 2005.

[59] W.-C. Feng, B. Code, E. Kaiser, M. Shea, W.-C. Feng, and L. Bavoil, "Panoptes: scalable low-power video sensor networking technologies," in *Proceedings of the 11th ACM International Multimedia Conference and Exhibition (MM '03)*, pp. 562–571, Berkeley, Calif, USA, November 2003.

[60] C. B. Margi, R. Manduchi, and K. Obraczka, "Energy consumption tradeoffs in visual sensor networks," in *Proceedings of 24th Brazilian Symposium on Computer Networks (SBRC '06)*, 2006.

[61] Crossbow Stargate platform, http://www.xbow.com.

[62] D. Jung, T. Teixeira, A. Barton-Sweeney, and A. Savvides, "Model-based design exploration of wireless sensor node lifetimes," in *Proceedings of the 4th European Conference on Wireless Sensor Networks*, pp. 277–292, 2007.

[63] L. Nachman, "New Tinyos platforms panel: iMote2," in *Proceedings of the Second International TinyOS Technology Exchange*, 2005.

[64] L. Ferrigno, S. Marano, V. Paciello, and A. Pietrosanto, "Balancing computational and transmission power consumption in wireless image sensor networks," in *Poceedings of the IEEE International Conference onVirtual Environments, Human-Computer Interfaces, and Measurement Systems (VECIMS '05)*, pp. 61–66, 2005.

[65] "Mica2 wireless sensor node," http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA2_Datasheet.pdf.

[66] R. Kleihorst, B. Schueler, A. Danilin, and M. Heijligers, "Smart camera mote with high performance vision system," in *Proceedings of ACM SenSys Workshop on Distributed Smart Cameras (DSC '06)*, 2006.

[67] A. Rowe, A. Goode, D. Goel, and I. Nourbakhsh, "CMUcam3: an open programmable embedded vision sensor," Tech. Rep. RI-TR-07-13, Carnegie Mellon Robotics Institute, 2007.

[68] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu, "SensEye: a multi-tier camera sensor network," in *Proceedings of the ACM Multimedia*, 2005.

[69] "Crossbow wireless sensor platform," http://www.xbow.com/Products/wproductsoverview.aspx.

[70] A. Rowe, D. Goel, and R. Rajkumar, "FireFly Mosaic: a vision-enabled wireless sensor networking system," in *Proceedings of 28th IEEE International Real-Time Systems Symposium (RTSS '07)*, 2007.

[71] A. Rowe, R. Mangharam, and R. Rajkumar, "RT-Link: a time synchronized link protocol for energy constrained multi-hop wireless networks," in *Proceedings of IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '06)*, 2006.

[72] M. M. Molla and S. I. Ahamed, "A survey of middleware for sensor network and challenges," in *Proceedings of the International Conference on Parallel Processing Workshops*, pp. 223–228, 2006.

[73] B. Rinner, M. Jovanovic, and M. Quaritsch, "Embedded midddleware on distributed smart cameras," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '07)*, vol. 4, pp. 1381–1384, Honolulu, Hawaii, USA, April 2007.

[74] H. Detmold, A. Dick, K. Falkner, D. S. Munro, A. van den Hengel, and R. Morrison, "Middleware for video surveillance networks," in *Proceedings of the Middleware for Sensor Networks (MidSens '06)*, pp. 31–36, 2006.

[75] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, vol. 21, no. 5, pp. 80–94, 2004.

[76] S. Meyer and A. Rakotonirainy, "A survey of research on contextaware homes," in *Proceedings of the Australasian Information Security Workshop Conference on ACSW Frontiers*, Australian Computer Society, Inc., 2003.