Security and Privacy Protection in Visual Sensor Networks: A Survey

THOMAS WINKLER and BERNHARD RINNER, Alpen-Adria Universität Klagenfurt
and Lakeside Labs

Visual sensor networks (VSNs) are receiving a lot of attention in research, and at the same time, commercial applications are starting to emerge. VSN devices come with image sensors, adequate processing power, and memory. They use wireless communication interfaces to collaborate and jointly solve tasks such as tracking persons within the network. VSNs are expected to replace not only many traditional, closed-circuit surveillance systems but also to enable emerging applications in scenarios such as elderly care, home monitoring, or entertainment. In all of these applications, VSNs monitor a potentially large group of people and record sensitive image data that might contain identities of persons, their behavior, interaction patterns, or personal preferences. These intimate details can be easily abused, for example, to derive personal profiles.

The highly sensitive nature of images makes security and privacy in VSNs even more important than in most other sensor and data networks. However, the direct use of security techniques developed for related domains might be misleading due to the different requirements and design challenges. This is especially true for aspects such as data confidentiality and privacy protection against insiders, generating awareness among monitored people, and giving trustworthy feedback about recorded personal data—all of these aspects go beyond what is typically required in other applications.

In this survey, we present an overview of the characteristics of VSN applications, the involved security threats and attack scenarios, and the major security challenges. A central contribution of this survey is our classification of VSN security aspects into data-centric, node-centric, network-centric, and user-centric security. We identify and discuss the individual security requirements and present a profound overview of related work for each class. We then discuss privacy protection techniques and identify recent trends in VSN security and privacy. A discussion of open research issues concludes this survey.

Categories and Subject Descriptors: K.6.5 [**Management of Computing and Information Systems**]: Security and Protection; K.4.1 [**Computer and Society**]: Public Policy Issues

General Terms: Security, Design, Human Factors

Additional Key Words and Phrases: Visual sensor networks, privacy protection, image sensor security

ACM Reference Format:
Thomas Winkler and Bernhard Rinner. 2014. Security and privacy protection in visual sensor networks: A survey. ACM Comput. Surv. 47, 1, Article 2 (April 2014), 42 pages.
DOI: http://dx.doi.org/10.1145/2545883

1. INTRODUCTION AND MOTIVATION

No matter whether along roads and highways [Bramberger et al. 2004; Farmer and Mann 2003], in sports stadiums, in shopping malls [Helten and Fischer 2004; Krempl and Wilkens 2011], in banks, at airports, or in underground stations [Ney and Pichler 2002], visual sensor networks (VSNs) have become a part of our daily life. Video surveillance [Cavoukian 2013b] is arguably one of the most widespread and well-known use

Authors' addresses: T. Winkler and B. Rinner, Alpen-Adria Universität Klagenfurt, Institute of Networked and Embedded Systems, and Lakeside Labs, Lakeside Park B02b, 9020 Klagenfurt, Austria; email: {thomas.winkler, bernhard.rinner}@aau.at.

DOI: http://dx.doi.org/10.1145/2545883

cases of VSNs; however, it is by far not the only one. Other VSN applications include environmental monitoring, smart homes and meeting rooms, entertainment, and virtual reality, as well as elderly care and assisted living. What all of these applications have in common is that visual sensors capture images that potentially reveal sensitive information about individuals, such as their identities or interaction patterns. Although privacy protection is a critical issue, in many applications, more general security requirements, such as integrity, authenticity, and timestamping for videos, must be considered. In a holistic approach, security and privacy protection must not stop at the data level. Any application-level protection approach will fail if the underlying infrastructure, such as the sensor node or the communication network, remain vulnerable. Naturally, similar security and privacy considerations are also valid for different image sensor types, such as infrared or thermal sensors.

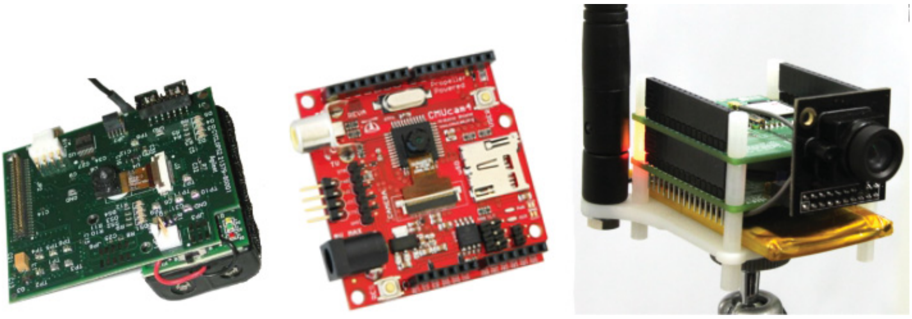
Privacy and security requirements for VSNs stand out from other applications because video data are easily analyzed by humans. By nature, images reveal much more than just the obvious identity information. They include subtle clues about people's habits, preferences, or social links. Humans are perfectly trained to grasp and process this type of information. Therefore, protection of data against insiders, such as system operators, is especially important in VSNs. To foster public acceptance of VSNs, it is crucial to be transparent about implemented security and privacy protection mechanisms. User-centric security mechanisms are important to reduce reservations about VSNs in public as well as private environments.

1.1. Characteristics of Visual Sensor Networks

VSNs [Soro and Heinzelman 2009] share many properties, techniques, and protocols with wireless sensor networks (WSNs). Several researchers view VSNs as a convergence between distributed smart camera networks [Wolf et al. 2003; Rinner and Wolf 2008] and WSNs [Akyildiz et al. 2007; Seema and Reisslein 2011]. The major differences between VSNs and WSNs is the amount of data obtained from the image sensor compared to scalar values read from, for example, temperature or humidity sensors in the case of WSNs.

The substantially larger amount of captured data has implications for many of the components of a VSN. The computing power of the node's processor has to be able to keep up with the amount of captured images. Early attempts for VSN devices employed the same low-performance 8-bit processors commonly found in WSNs. The Cyclops (see Figure 1(a)) by Rahimi et al. [2005] uses a Atmega128 microcontroller that runs at 7.3MHz and offers 64kB RAM. Computer vision on such a resource-constraint system is extremely challenging. More recent designs make use of more capable 32-bit processors. The CMUcam4 [Agyeman and Rowe 2012] (see Figure 1(b)) or the even more powerful Citirc [Chen et al. 2008] platform are two examples of this trend. Citric is based on an ARM CPU clocked at 624MHz and comes with 64MB of RAM. Other recent VSN devices support even higher performance by integrating multicore processors running at clock frequencies between 1 and 1.5GHz. A platform specifically designed for secure and privacy-preserving VSN applications is TrustEYE.M4 [Winkler and Rinner 2013] (see Figure 1(c)). It is based on an ARM Cortex M4 processor clocked at 168MHz, 4MB of external SRAM, a dedicated hardware security module, and an OmniVision OV5642 image sensor. For wireless connectivity, a WiFi extension board can be attached. TrustEYE.M4 can be used either in standalone mode or as a secure sensing unit in a larger camera device.

The differences between WSN and VSN do not stop at data processing. A central idea of VSNs is to keep data processing local to reduce the amount of transmitted data. Ideally, only event information is transmitted by VSN devices. However, for verification purposes or for the detailed assessment of critical situations, it is often desirable to



(a) Cyclops [Rahimi et al. 2005] node is based on a 7.3MHz Atmega128 and has 64kB RAM.

(b) CMUcam4 [Agyeman and Rowe 2012] uses a Parallax P8X32A processor clocked at up to 80MHz and has 32kB of internal RAM.

(c) TrustEYE.M4 [Winkler and Rinner 2013] is based on an ARM Cortex M4 CPU clocked and 168MHz and has 4MB of SRAM Wireless connectivity is provided via a WiFi extension board.

Fig. 1. Examples of VSN devices with substantially different performance and capabilities.

deliver live video footage as well. In contrast to WSNs, the wireless communication interfaces of VSN devices must therefore support the transmission of high-volume data. Additionally, also related protocols such as MAC or routing must be adapted and designed to meet these requirements.

Depending on the computing capabilities of VSNs, different types of security solutions can be deployed. For low-performance devices such as Cyclops [Rahimi et al. 2005], standard asymmetric cryptography is typically not suitable. Therefore, a number of dedicated and lightweight security techniques and protocols for WSNs have been proposed [Perrig et al. 2002; Karlof et al. 2004; Zhu et al. 2006; Chen et al. 2009] that are also suitable for VSNs. More powerful devices such as Citric [Chen et al. 2008] are capable of running state-of-the-art security solutions as found on smartphones or desktop computers. In this survey, we intentionally focus on fundamental security and privacy requirements that apply uniformly to all types of VSN systems. Implementation and system-specific aspects are also covered but are not the primary focus.

Other important aspects of VSNs aside from lightweight, power-efficient hardware platforms are collaboration and in-network processing. A single VSN device has only a limited field of view, but VSNs are typically designed to cover large areas. Therefore, multiple spatially distributed nodes are required. To avoid centralized control and data processing, VSNs use peer-to-peer communication for coordination, configuration, data exchange, handover of tracked objects, or data fusion. To simplify deployment of spatially distributed VSNs, they rely no longer only on dedicated communication networks but make use of existing infrastructure that is not under full control of the VSN operators. Wireless communication is used where installation of wired networks would be too costly or cumbersome. Open networks and wireless communication make VSNs much more vulnerable than traditional closed-circuit surveillance camera networks.

The limited resources of VSN nodes are another reason for in-network processing. Solving complex tasks often requires collaboration from adjacent nodes with free resources. Cooperative processing allows the analysis of collected information locally instead of transmitting raw data over multiple hops to a central sink. Local processing saves energy, which is a scarce resource in sensor networks. From a security perspective, internode collaboration raises a number of challenges, including the distribution of cryptographic keys in multi- and broadcast scenarios, secure discovery and localization of adjacent nodes, secure MAC and routing, or trustworthy data sharing and fusion.

1.2. Application Scenarios

To illustrate the different security and privacy requirements for VSNs, we discuss three typical application scenarios. In this survey, we focus on VSNs designed and deployed for specific applications. Participatory sensing applications or content generated by user devices (e.g., smartphones) are not considered. The individual security aspects mentioned in the following scenarios are discussed in detail in Sections 2 through 7.

Reactive Monitoring for Enforcement. A well-established domain of VSNs are enforcement applications where they are used to collect evidence of law violations. A prevalent scenario is traffic monitoring [Cucchiara et al. 2000; Bramberger et al. 2004; Arth et al. 2006]. Highways, roads, or intersections are monitored to detect traffic law violations such as tailgating, speeding, or illegal parking. Many systems are *reactive*, which means that images or videos are only recorded if a critical event has occurred. Recording is often triggered by additional sensors, such as radar or laser range finders. However, on-board event detection by analyzing the steady image stream becomes more and more common [Bischof et al. 2010]. Collected evidence must be trustworthy in case of a dispute at court. This property is also referred to as *nonrepudiation* of the evidence. Nonrepudiation requires guarantees that the evidence was not manipulated after collection (*integrity*), that it was collected by a specific camera (*authenticity*) with known and trustworthy properties, and finally that the evidence was captured at a specific point in time (*timestamping*). These properties must be tightly bound to the evidence to achieve nonrepudiation.

Reactive Monitoring for Private Safety. VSNs are used also in private environments such as elderly care and assisted living [Aghajan et al. 2007; Fleck and Straßer 2008; Bamis et al. 2010; Pinto 2011]. In these scenarios, people voluntarily give up a certain amount of personal privacy in exchange for services. These services are typically reactive, which means that inhabitants are monitored but data are delivered to a monitoring facility only if an unusual event was detected. Nonrepudiation is not a requirement, because the collected information is usually not meant to be used as evidence at court.

When agreeing to the terms of a home monitoring system, participants accept that personal data are made available to a limited group of persons (i.e., operating personnel) under certain circumstances. It must be ensured that access to personal data is reliably limited to this legitimate group (*access authorization*) and that data are protected while being transmitted to the monitoring facility (*confidentiality*). Both the software as well as the hardware of the installed VSN devices have to be protected against attacks by outsiders to ensure that the devices cannot be abused by people in the neighborhood to, for example, illegitimately obtain a video feed of other people's private environments.

Proactive Monitoring for Public Safety. VSNs are not always used with such a strong focus as in enforcement and private safety applications. Large networks of cameras are deployed in cities and public places such as train stations, airports, or shopping malls [Ney and Pichler 2002; Hampapur et al. 2007; Hampapur 2008; Bulkeley 2009]. The installed cameras are used primarily for monitoring purposes or as a deterrent [Norris 2009]. These applications are commonly subsumed under the relatively vague term *public safety*. In many of these systems, data are captured *proactively* and transmitted continuously to a central monitoring and archiving facility. Captured videos contain the *identities* of all persons in the field of view of the camera even though their *behavior* would be sufficient for most safety applications. Collecting identities and the ability to track individuals over large distances clearly make proactive and large-scale VSNs intrusive to people's *privacy*. Therefore, data *confidentiality* and *access authorization* are important requirements. Moreover, monitored people should be asked for *consent* to monitoring, and they should remain in *control* over their personal data. If an offense were recorded and the evidence used at court, nonrepudiation guarantees might be required for proactive monitoring systems.

1.3. Contribution and Outline

Security in the related field of WSNs has been studied by many researchers and various reviews [Wang et al. 2006; Zhou et al. 2008; Mpitziopoulos et al. 2009; Chen et al. 2009] have been published.

The motivation for this survey specifically on VSN security and privacy stems from two main observations. First, a primary issue with visual information is that it can be analyzed easily by nonexperts. Humans are trained to interpret and rely on visual information in their daily lives and their capabilities in this area go beyond those of state-of-the-art computer vision systems. Consequently, protection is required not only against outside attackers but it also must be ensured that insiders get only limited access to sensitive information. The second motivation is the lack of a systematic review of VSN security and privacy. We present a classification scheme based on four different classes: (1) security and privacy protection for collected data; (2) security concerns of monitored people and the requirement for providing transparency about the purpose, the tasks, and security properties of VSNs; (3) device security; and finally (4) network security. Based on our classification scheme, we give a profound overview of existing work. From the analysis of the state of the art, we derive prospective future research topics and open challenges. The audience of this survey is not limited to security experts. We describe VSN security and privacy in a way that makes these topics accessible to the sensor network, computer vision, and embedded systems communities.

The remainder of this survey is organized as follows. In Section 2, we discuss the threats, attack scenarios, and challenges in VSN security. We also give an overview of VSN security requirements. The subsequent sections individually cover the major VSN security topics in detail and present involved requirements and related work. These topics are data-centric security (Section 3), privacy (Section 4), user-centric security (Section 5), node-centric security (Section 6), and network-centric security (Section 7). Finally, Section 8 summarizes our observations on the current state and trends in VSN security, outlines open research questions, and concludes the survey.

2. THREATS, CHALLENGES, AND REQUIREMENTS FOR VSN SECURITY

In this section, we define the threats and attack scenarios that have to be faced by designers of VSNs. We give an overview of the involved challenges and present a classification of the core VSN security requirements. This classification is shown later in Figure 2 and serves as a central reference point for the detailed discussion of VSN security and privacy aspects and the related work throughout Sections 3 through 7.

2.1. Threats and Attack Scenarios

Attacks on VSNs can be classified based on their goals, by whom they are performed, and the level at which the attack is carried out. We distinguish passive attacks that aim to illegitimately collect data from a VSN; active attacks where (partial) control over the VSN infrastructure is achieved; and finally, attacks that aim at disrupting the services provided by VSNs. Furthermore, we discuss threats from outsiders and insiders as well as hardware- and software-based attacks.

Illegitimate Data Access. In this scenario, an attacker is interested in eavesdropping the information that is exchanged in the VSN. The goal of the attacker is to

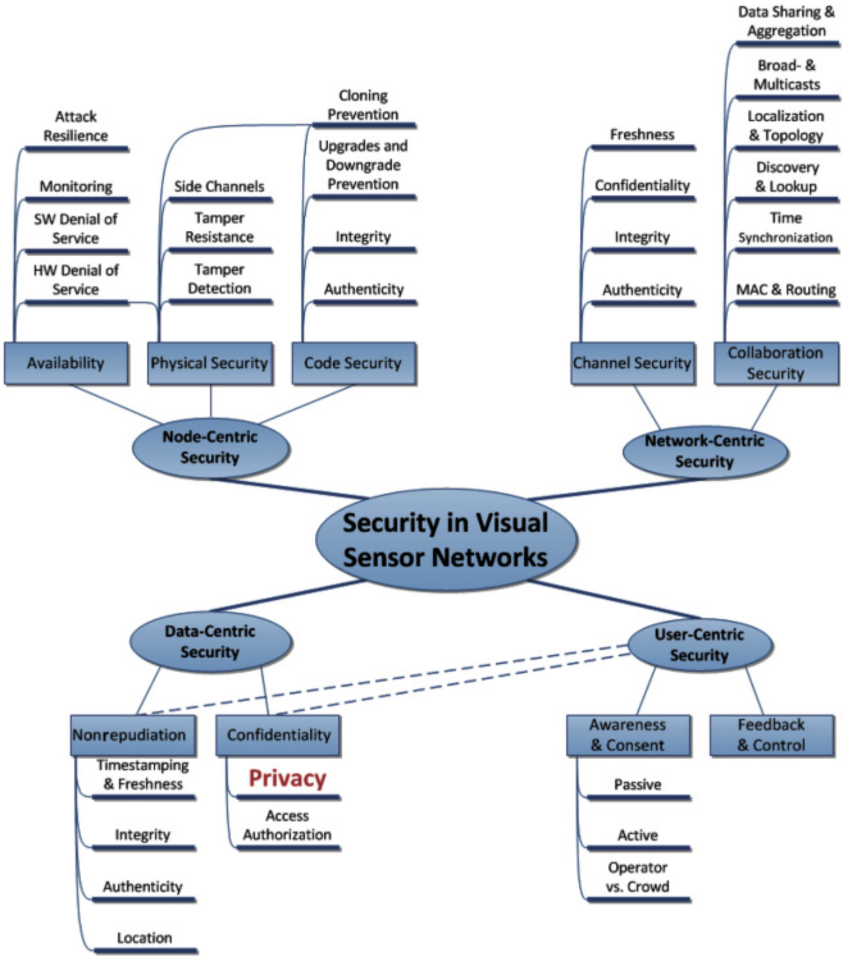


Fig. 2. Our classification scheme for VSN security requirements consists of four areas. Data-centric security focuses on nonrepudiation and confidentiality for recorded data. Our data-security definition assumes that security guarantees have the same lifetime as the data. Node-centric security means all aspects directly related to the VSN device including both its software and its hardware. Network-centric security addresses 1:1 and 1:n communication. In contrast to data-centric security, network-centric security guarantees are only valid during transmission. User-centric security means making monitored people aware of VSNs and giving them the possibility to check if and how their personal data are protected. From the perspective of users, data confidentiality and nonrepudiation are also highly important with is illustrated by the dashed lines.

use the information for her or his own purposes, such as scouting a certain region of interest while remaining undetected. Consequently, this type of attack is usually passive and is performed by, for example, overhearing the communication channel.

Illegitimate Control. In this scenario, the attacker is no longer passive but takes active measures to achieve (partial) control over the network. Injection of forged control messages could allow an attacker to reposition a PTZ camera to cover an area that is of interest for the attacker or, vice versa, make sure that certain areas are not covered by the system. To exercise control over the network, it might be insufficient to just forge and inject control messages, but an attacker might need to capture and compromise one of the nodes of the VSN. This allows the issuance of apparently valid commands and requests and to launch attacks from inside the network.

Service Degradation and Denial of Service. An attacker is not primarily interested in obtaining data from the VSN or gaining control over the network. The main goal is to reduce the availability and utility of the VSN such that its legitimate users can no longer rely on the services usually provided by the VSN. This can be achieved, for example, by jamming of wireless communication, by injection of useless requests at a rate that overwhelms the VSN nodes, or by manipulating the routing information used for multihop data communication. Capturing and compromising nodes as described in previous attack scenarios potentially simplifies denial of service (DoS) attacks, because attacks from inside the network might not have been taken into account by the designers of the VSN. Last but not least, physical attacks on selected VSN nodes can result in severe service degradation of the entire network.

Threats from Outsiders versus Insiders. Attacks by outsiders can be mitigated by appropriate protection methods, such as those based on data encryption, message authentication, or digital signatures. Insiders are system operators or security guards who have legitimate access to the VSN's control and management facilities. Insiders can misuse their access privileges to disrupt the services of the camera network, such as by intentionally introducing network configuration errors. These DoS attacks by insiders can be mitigated by designing the system such that the four eyes principle is enforced for all configuration changes to the control and management infrastructure. As part of fulfilling standard monitoring duties, insiders usually require access to video and image data delivered by the cameras. That means that the same protection mechanisms, such as data encryption, cannot be used as those used against outsiders. Legitimate operators require at least partial access to unprotected video data to fulfill their duties. It must be ensured that only the minimal required amount of information is disclosed. For most monitoring purposes, behavioral information is sufficient while identities can be hidden. Moreover, technical precautions must be taken such that legitimate users cannot disclose any information to outsiders.

Software versus Hardware Attacks. By software attacks, we mean all types of manipulation that target the software stack of a VSN node. These attacks are performed typically from remote via the wired or wireless communication channels of VSNs. Attacks include not only the modification of existing software and the installation of new software (e.g., malware, root kits) but also attacks on routing and MAC protocols. Many mitigation strategies for software attacks have been proposed in the literature, but a fundamental problem is that pure software security solutions themselves are vulnerable to software attacks. As a consequence, various forms of hardware security support have been proposed, such as smartcards [Dietrich and Winter 2009, 2010]; CPU instruction set extensions [Winter 2008; ARM Limited 2009]; or dedicated security ASICs, such as Trusted Platform Modules (TPMs) [Trusted Computing Group 2011; Martin 2008]. Although hardware support can substantially increase the overall security of an embedded system, it provides only limited protection against physical attacks on VSN nodes such as node capture and hardware tampering. Prevention of theses attacks is inherently difficult, but at least detection mechanisms should be incorporated into VSN devices. More sophisticated hardware attacks involve the reverse engineering of integrated circuits or the exploitation of side channels where information is leaked, for example, via the power consumption patterns of individual microchips or entire devices.

2.2. Major Design Challenges

Subsequently, we discuss the most relevant challenges in VSN security and privacy together with the involved trade-offs.

Open System Architecture. Modern VSN devices are often designed to make use of existing communication infrastructure such as WiFi networks or the Internet. This is a major difference to traditional closed-circuit networks, which are under full control of the system operator. The use of open infrastructure is a challenge not only for protecting sensitive data that is transmitted. Remote attacks on VSN nodes are also easier if they are connected to open networks.

Limited System Resources. Low power consumption, small size, and affordable price are important design goals for VSN devices. A direct consequence is that the amount of processing power and the available memory are limited. The challenge is that to a large extent, the system's resources are consumed by the on-board image processing and analysis applications, and typically only a small amount is left for security and privacy protection. In practice, a suitable trade-off between system performance and the implemented security functionality must be found. Ideally, security and privacy protection should have only minimal impact on system performance.

Limited Physical Control. Nodes of a VSN are deployed in public environments where they are not under full control of the owners and operators. Nodes are mounted on walls or poles where attackers can easily access them. Simple attacks are node destruction or node theft. Advanced attacks are node capture, where an attacker obtains physical access to a node to, for example, extract data from the device. Extracted data could be cryptographic keys or sensitive image data temporarily stored on the node for later processing or transmission. Preferably, all sensitive data should be stored in on-chip memory, where it is protected against basic hardware attackers. However, due to cost reasons, on-chip memory is usually very small, and external memory has to be attached to the processing core. One approach is to encrypt all data stored in off-chip memory, which, however, makes memory access more expensive in terms of computing power.

Visual Data Privacy. Images and videos can be easily interpreted by humans and potentially reveal much more information than data captured with other sensors. This aspect makes privacy protection in VSNs even more challenging than in other application domains. For real-world deployments, appropriate trade-offs within this privacy protection and system utility design space (see Section 4.1) must be identified. It must be ensured that the system remains usable for its intended purpose while only the minimally required amount of information is disclosed to system operators.

2.3. Security Requirements

Figure 2 presents an overview of the requirements and our approach of partitioning them into four groups. The first group addresses data-centric security aspects (see Section 3), which are primarily *nonrepudiation* and *confidentiality*. *Privacy* (see Section 4) is defined as a subset of confidentiality and denotes protection of sensitive data against insiders. The second area is user-centric security (see Section 5), which covers requirements of persons monitored by the VSN.[1] These requirements include user *consent* to monitoring as well as remaining in *control* over personal data. The third group deals with security requirements of a single sensor node (see Section 6). It includes *availability* of the node and its services, *physical* security of the device, and security of the *code* that is executed on the device. Finally, we extend our considerations from the node level to the network (see Section 7). We address the communication

[1] In this work, we use the term *users* synonymously with people who are monitored and recorded by the VSN.

channel between two VSN devices as well as security in larger, *collaboration*-oriented VSN systems.

The four groups shown in Figure 2 also illustrate the different scopes when dealing with security. Data-centric security requirements apply directly to captured images and all types of derived data. Security in this context is tightly tied to the data, which means that security properties remain attached to the data for the data's entire lifetime, starting with its creation, including transmission and usage, and finally storage and long-term archiving. This is in contrast to the scope of certain network security aspects. Channel-oriented security mechanisms such as the Secure Sockets Layer (SSL) ensure data integrity, authenticity, and confidentiality for data transmitted from source to destination. These properties of the channel allow the user to assess that data were properly protected during transmission, but no statements regarding security of the data can be made for the time before and after transmission. Consequently, data security as defined in our model is a significantly stronger property than channel-oriented network security. What becomes apparent from Figure 2 is that certain security requirements are (partially) redundant. Channel-oriented network security might not be required if protection is already applied previously at the data level. The major difference in this case is the lifetime of the protection. Which security approach and what lifetime properties are required cannot be defined per se but depend on the application.

3. DATA-CENTRIC SECURITY

Data-centric security addresses the protection of all data made available by a camera system. The definition of data in this context is not limited to raw images but also includes processed image data, all types of derived information, and high-level event descriptions. For all delivered data, *nonrepudiation* as well as *confidentiality* must be ensured. In our classification, data-centric security properties are tightly bound to the data and have the same lifetime as the data.

3.1. Data-Centric Security Requirements

3.1.1. Nonrepudiation. Nonrepudiation subsumes requirements that are important to answer by *whom* (i.e., by which camera), *where*, and *when* data were produced. Additionally, it must be ensured that any data *manipulation* is detected.

Authenticity. In many applications, it is important to know by *whom* data were produced. In VSNs, this is equivalent to knowing the identity of the camera that captured and processed a video stream. This can be achieved by explicitly authenticating the nodes of a VSN and embedding this information into the video streams. Authenticity information can be digital signatures or watermarks. Alternative authentication approaches rely on forensics techniques and use the sensor-specific noise patterns to identify the origin of images. An aspect that goes beyond basic authenticity is the inclusion of device status information as part of the authenticity information. To assess the overall trustworthiness of received data, it is important to know which software was running on the VSN device at the time the image was captured and processed.

Location. The reasons why evidence is required *where* images or videos were captured are similar to those mentioned previously for authenticity. For example, if image data will be used at court, it must not be disputable where the data were captured. Location information can be collected from dedicated positioning devices (e.g., GPS receivers) that are part of the VSN nodes. In case of static installations, it might be sufficient to rely on predefined location information or to associate a location with the identity of a device.

Timestamping and Freshness. To prevent replay attacks where recorded videos are injected into the network to replace the live video stream, freshness of image data must be guaranteed. For basic freshness guarantees, no real-time clock on the VSN node is required. However, if evidence is required *when* an image was taken, it must be timestamped. For this purpose, a reliable time source is required. The timestamp is bound to the data such that every manipulation of the timestamp is detected. Timestamping of images answers not only the question of when an image was taken but at the same time also satisfies the requirement for image freshness guarantees.

Integrity. Data coming from a camera can be intentionally modified by an attacker during transmission or when stored in a database. Data integrity is ensured by using checksums, digital signatures, and watermarks. Integrity protection must cover not only the payload data but also the attached data such as location information and timestamps. Often overlooked is that integrity protection is important not only for single frames but also for video sequences. Simple reordering of images can substantially change a video's meaning.

3.1.2. Confidentiality. Confidentiality denotes the protection of images, videos, and all derived data against access by external parties. Confidentiality must be maintained throughout the entire lifetime of the data, from image capturing to long-term archiving in a database. Confidentiality is typically achieved via data encryption. Internal parties such as system operators or security guards require access to confidential information to fulfill their duties.

Access Authorization. Access to confidential image data must be limited to a group of legitimate system users, such as security guards. An access authorization scheme must ensure that only persons with adequate security clearance get access to video data. For access to especially sensitive data, involvement of more than one operator should be required to prevent misuse. If a video stream contains different levels of information (e.g., full video, annotations), access should be managed separately for each level. Additionally, all attempts to access confidential data should be securely logged.

Privacy. In our classification (see Figure 2), privacy is a subproperty of confidentiality. Whereas confidentiality denotes protection of all data against external parties, privacy means protection of sensitive data against misuse by legitimate users (i.e., insiders). For system operators who perform monitoring tasks, behavioral information is usually sufficient and identity information is not required. This can be achieved by automatic detection and removal of sensitive image regions such as people's faces. Since privacy is an extremely important but complex aspect of VSNs, Section 4 is entirely dedicated to privacy protection.

3.2. Related Work on Data-Centric Security

Serpanos and Papalambrou [2008] provide an extensive introduction to security issues in the domain of smart cameras. They discuss the need for confidentiality, integrity, freshness, and authenticity for data exchanged between cameras. Embedded systems might not have sufficient computing power to protect all data using cryptography. In such a situation, the authors propose to concentrate on the most important data. This work recognizes the overlap of confidentiality and privacy protection and emphasizes the importance of data protection not only against external attackers but also against legitimate system operators. Senior et al. [2005] discuss critical aspects of a secure surveillance system, including what data are available and in what form, who has access to data, and in what form and how long the data are stored. Data confidentiality is ensured via encrypted communication channels. User privacy is a major concern, and it is suggested that videos are analyzed and sensitive regions are re-rendered. The resulting, multiple video streams contain different levels of data abstraction and are separately encrypted. Video analysis, processing, and encryption could either be done directly on the cameras or via a dedicated privacy console. Schaffer and Schartner [2007] present a distributed approach to ensure confidentiality in a video surveillance system. They propose that the video stream is encrypted using a hybrid cryptosystem. Encryption is performed for full video frames without differentiating between sensitive and nonsensitive image regions. A single system operator is not able to decrypt a video, but multiple operators have to cooperate. This property is achieved by the fact that every operator is in possession of only a part of the decryption key. These key shares are stored in smartcards and are used in a multiparty computation to decrypt the video.

Integrity protection of image and video can be achieved by means of, for example, hash functions together with digital signatures or by embedding watermarks into the video content. Regardless of the chosen approach, an important design decision is whether the integrity protection technique is tolerant toward certain, acceptable image modifications or not. The work of Friedman [1993] aims at "restoring credibility of photographic images" and therefore does not accept any image modifications. Specifically, authenticity and integrity of images taken with a digital still camera should be ensured. This is achieved by extending the camera's embedded microprocessor with a unique, private signature key. This key is used to sign images before they are stored on mass storage. The public key required for verification is assumed to be made available by the camera manufacturer. This work is one of the earliest approaches toward a trustworthy, digital imaging system. Similar systems, deployed in Canon and Nikon DSLR cameras, have been compromised [Sklyarov 2010; Katalov 2011]. The major problem is that the signature key is not properly protected and can be extracted from the camera's firmware. Even worse, the signature key is not unique but shared for all cameras of the same model. Quisquater et al. [1997] propose an approach for integrity protection and authentication for digital video stored on tape in the DV format. They use SHA-1 to compute the hashes of the images. To be less sensitive to transmission or tape errors, the authors suggest that the images are divided into blocks that are hashed separately. Authenticity is ensured by signing the hash values. The hash of the previous image is also included in the signature to maintain correct ordering of video frames.

Atrey et al. [2004, 2006] present a concept to verify the integrity of video data. In their work, they differentiate between actual tampering and benign image modifications. Operations that do not change the video semantically, such as image enhancements or compression, are defined as acceptable. Tampering of video data is divided into spatial and temporal modifications. Spatial tampering includes content cropping as well as removal or addition of information. Temporal tampering refers to dropping or reordering of frames that might result from, for example, network congestion. The authors argue that temporal tampering is acceptable as long as the semantic meaning of the video is not substantially affected. They propose a configurable, hierarchical secret sharing approach that is shown to be tolerant to benign image modifications while tampering is detected. He et al. [2004] discuss the design of a video data integrity and authenticity protection system that does not operate on frames but on objects. Objects are separated from the video background using segmentation techniques. An advantage is that network bandwidth can be saved by transmitting primarily object data, whereas background data are updated less frequently. The integrity protection system is designed to tolerate certain modifications such as scaling, translation, or rotation. Considering these requirements, appropriate features are extracted from the detected objects as well as the background. A hash of these features together with error correction codes is embedded into the video stream as a digital watermark.

Digital watermarks are a popular technique to secure digital media content. A watermark is a signal that is embedded into digital data that can later be detected, extracted, and analyzed by a verifier. According to Memon and Wong [1998], a watermark can serve several different purposes. One purpose can be proof of ownership where a private key is used to generate the watermark. Other applications are authentication and integrity protection, usage control, and content protection. Depending on the application domain, watermarks can be visible or invisible. When used for integrity protection, watermarks have the advantage that they can be designed such that they are robust against certain image modifications, such as scaling or compression [Albanesi et al. 2001; Bartolini et al. 2001]. An example where watermarking is used as part of a digital rights management (DRM) system for a secure, embedded camera is presented by Mohanty [2009]. He describes a secure digital camera system that provides integrity, authenticity, and ownership guarantees for digital video. This is achieved using a combination of watermarking and encryption techniques. Due to the high computational effort, a custom hardware prototype based on an FPGA is used to meet the real-time requirements.

Another approach to verifying image integrity and authenticity is based on image forensics techniques. Digital image sensors are not perfect, and a certain amount noise is produced when capturing images. This sensor noise can be exploited to assert the integrity and authenticity of digital images [Chen et al. 2008; Li 2010; Sutcu et al. 2007].

In our own work [Winkler and Rinner 2011], we rely on hardware-based security techniques to implement integrity, authenticity, freshness/timestamping, and strong confidentiality. All image data delivered by the TrustCAM prototype system is digitally signed and encrypted. The used RSA keys are protected by the camera's TPM chip, which additionally provides a unique platform identity.

Table I presents a comparison of the discussed approaches for data-centric protection. The comparison matrix makes it very clear that researchers have put an emphasis on integrity and authenticity of video and image data. This strong focus can be explained by the wide use of video surveillance in enforcement applications and the desire to provide indisputable evidence. Table I also shows that confidentiality has to go hand in hand with the access authorization. Freshness and timestamping, as well as the location of data capturing, are less frequently addressed by the reviewed literature.

4. PRIVACY

Cameras allow the field of view of observers to be extended into areas where they are not physically present. This "virtual presence" of an observer is not necessarily noticed by monitored persons. In the resulting but misleading feeling of privacy, persons might act differently than they would in the obvious presence of other people. This example makes it apparent that privacy in video surveillance is an issue that needs special consideration. But when trying to identify what forms of privacy protection are appropriate, the picture becomes less clear. One reason is that there is no common definition of privacy. As discussed by Moncrieff et al. [2009] and Senior et al. [2005], the notion of privacy is highly subjective, and what is acceptable depends on the individual person as well as cultural attitudes.

The problem of protecting an area against capture by cameras is addressed by Truong et al. [2005]. In their capture-resistive environment, camera phones are prevented from taking images. Emitted IR light is retroreflected by the mobile phone's image sensor. These reflections are detected by the system and used to localize the mobile phone, which is then neutralized by intense, directed light emitted by a video beamer.

Table I. Comparison of Implementations of Data-Centric Security

	Integrity	Authenticity	Freshness / Timestamping / Ordering	Confidentiality	Access Authorization	Location
Albanesi et al. [2001]	●	●	○	○	○	○
Atrey et al. [2004], Atrey et al. [2006]	●	○	◐	○	○	○
Bartolini et al. [2001]	●	●	○	○	○	○
Chen et al. [2008]	●	●	○	○	○	○
Friedman [1993]	●	●	○	○	○	○
Li [2010]	●	●	○	○	○	○
Mohanty [2009]	●	●	○	○	○	○
He et al. [2004]	○	○	○	○	○	○
Memon and Wong [1998]	●	●	○	○	○	○
Quisquater et al. [1997]	●	●	○	○	○	○
Schaffer and Schartner [2007]	○	○	○	●	●	○
Senior et al. [2005]	○	○	○	●	●	○
Serpanos and Papalambrou [2008]	●	●	◐	●	●	◐
Sutcu et al. [2007]	●	●	○	○	○	○
Winkler and Rinner [2011]	●	●	●	●	●	○

White bullets represent unsupported, gray bullets partially realized, and black bullets fully covered properties.

Although this is an interesting approach to preserve privacy in selected areas, it is not practical for large deployments. Therefore, many researchers focus on the opposite approach, where cameras actively detect and protect privacy-sensitive image regions.

As pointed out by Cavallaro [2007] or Fidaleo et al. [2004], it is usually more important to be able to observe the behavior of a person than knowing the actual identity. This is achieved by identification and obfuscation of personally identifiable information such as people's faces [Chen et al. 2007; Martínez-Ponte et al. 2005]. Only in situations where, for example, a law was violated is this personal information of interest and should still be available to authorized parties. The main challenge of such an approach is to determine which image regions are actually sensitive. As Saini et al. [2010] argue, video data not only includes direct identifiers such as human faces but also quasi identifiers. These quasi identifiers are often based on contextual information and allow the ability to infer the identity of persons with a certain probability. Such basic contextual information about an event includes what happened, where it happened, and when it happened. Vagts et al. [2009, 2010] present an approach that addresses privacy protection not at the sensor level but at a higher abstraction level. As part of their task-oriented privacy enforcement system, data are collected only if required for a surveillance task. For that purpose, each task must be fully specified before data collection is started.

It must be noted that privacy considerations in this survey mainly revolve around the protection of visual, privacy-sensitive data, because protection requirements in this domain make VSNs stand out from other applications. In every deployment of a privacy-preserving VSN, considerations must go clearly beyond these aspects. Communication patterns between adjacent cameras, chronology of data exchanges, or the location of nodes might give away sufficient information that allows an observer

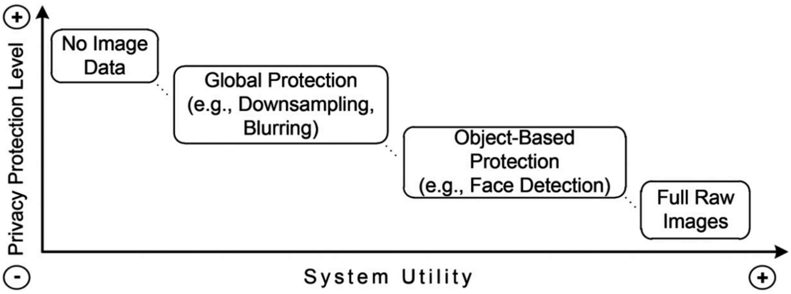

No Image Data
Global Protection (e.g., Downsampling, Blurring)
Object-Based Protection (e.g., Face Detection)
Full Raw Images
Privacy Protection Level
System Utility
+
-
+

Fig. 3. The privacy protection design space.

to derive privacy-sensitive information. Although visual information clearly is highly sensitive with respect to privacy of monitored people, such secondary information channels must not be overlooked in the design of a real-world system.

4.1. Privacy Protection Requirements and Techniques

In this survey, we define confidentiality as protection of all data delivered by a VSN against external parties. With privacy, we mean protection of sensitive data against insiders such as security guards who have legitimate access to the VSN. Anonymization can be seen as a primary method of privacy protection [Seys et al. 2001]. Consequently, the most common way of realizing privacy protection in video surveillance is visual anonymization. Anonymization is achieved by detection and protection of identity information based on computer vision techniques. Primary identifiers in the context of VSNs are human faces. The result of anonymization is that identities are hidden while behavioral information is preserved. Figure 3 illustrates that there is no single best approach for achieving privacy protection, but there is a design space with a variety of solutions with different advantages and disadvantages. Delivering no image data yields the best privacy protection but at the same time makes monitoring by system operators impossible. Providing full, raw images results in the best monitoring performance but entails a total loss of privacy. In between these two extremes, global and object-based techniques exist with different trade-offs between privacy protection and system utility. First, object-based approaches rely on the identification and protection of sensitive regions such as motion blobs, persons, or faces. Only the identified regions are protected, and the rest of the image remains visible. Second, global approaches apply uniform protection operations (e.g., downsampling, blurring, mosaicing, or edge detection) to the entire raw image and are therefore not prone to errors in the detection of sensitive regions. Global approaches are not yet very prominent in the related literature. A noteworthy exception is the approach by Saini et al. [2012]. Their work builds on the idea of global privacy protection approaches, but they additionally use the output of unreliable detectors to selectively adapt the applied protection. Compared to pure object-based approaches, the authors achieve higher robustness against inaccuracies of the detectors while maintaining an overall higher level of visual quality.

The subsequent paragraphs summarize commonly used object-based protection techniques. A requirement for object-based protection is the identification of privacy-sensitive image regions such as human faces or vehicle license plates. If this identification does not work reliably, privacy is at risk. A single frame of a sequence where sensitive regions are misdetected can break privacy protection for the entire sequence.

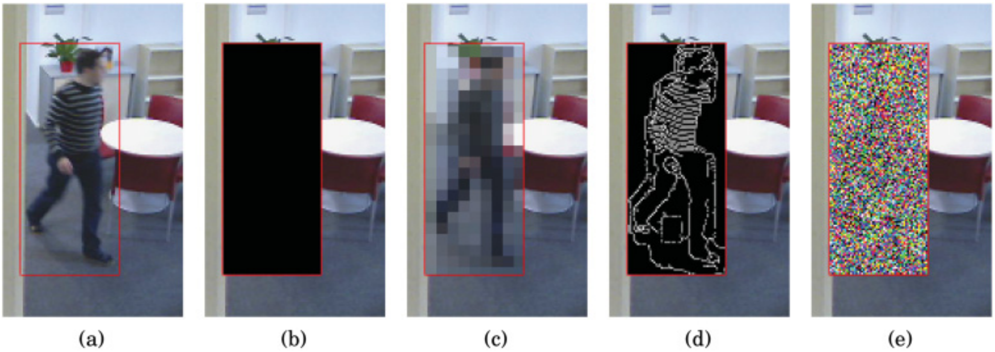


Fig. 4. Privacy is ensured by protecting especially sensitive image regions. In this example, privacy protection is applied for the entire region where the moving person was detected (Figure 4(a)). Illustrated protection techniques include blanking (Figure 4(b)), pixelation (Figure 4(c)) abstraction by means of edge detection (Figure 4(d)), and encryption (Figure 4.1(e)).

Figure 4 illustrates some of these techniques, including blanking, pixelation, abstraction, and encryption.

Blanking. One way to deal with sensitive image regions is to completely remove them from the image, leaving behind blanked areas. Although providing perfect privacy, the usefulness of the system is reduced, as not even basic behavior can be observed and identities of persons are lost. Only the presence and location of persons can be observed. Some researchers, such as Cheung et al. [2006], apply video inpainting techniques to fill the blank areas with background. This way, an observer can no longer notice that information was removed from the video.

Obfuscation and Scrambling. The purpose of obfuscation is to reduce the level of detail in sensitive image regions such that persons can no longer be identified yet their behavior remains perceptible. Researchers apply different techniques including mosaicing, pixelation, blurring [Chinomi et al. 2008; Wickramasuriya et al. 2004], warping [Korshunov and Ebrahimi 2013], cartooning [Erdélyi et al. 2013], or high and lossy compression. Another technique to protect sensitive image regions is scrambling. In its basic form, JPEG compressed images are obscured by pseudorandomly modifying the DCT coefficients [Dufaux and Ebrahimi 2006] of sensitive regions.

Abstraction. This popular technique replaces sensitive image regions with, for example, bounding boxes or, in case of persons, with avatars, stick figures, and silhouettes [Senior et al. 2005]. Another form of abstraction is metainformation attached to a video. This can be object properties such as position and dimensions, but also names of identified persons [Tansuriyavong and Hanaki 2001]. Depending on the type of abstraction, either behavior, identity, or both can be preserved. Note that if identity is preserved, additional protection (e.g., by encryption) should be considered.

Encryption. Data encryption is used by many systems to protect sensitive regions. When encrypted, regions of interest can no longer be viewed by persons who do not have the appropriate decryption keys. Simple encryption protects not only the identity of monitored persons but also their behavior. Upon decryption, both—identity and behavior—are revealed. By using multiple encryption keys or split keys as described by Schaffer and Schartner [2007], a system can be designed that requires cooperation among multiple operators to decrypt the original data, which provides some protection against operator misuse.

A privacy protection system can also support multiple privacy levels at the same time where a video stream contains several substreams with different types of information. Depending on their sensitivity, these levels have to be separately encrypted with one or more individual encryption keys. A multilevel approach allows a privacy protection system to be designed that presents different types of information to observers depending on their security clearance. Low-privileged operators can only access the version of the stream where behavioral data are visible, whereas supervisors or government agencies could get access to the original data that contain the identities of monitored persons. In our own work [Winkler and Rinner 2011], we have explored a multilevel approach where for especially sensitive video content, a combination of two or more encryption keys is used to enforce the four eyes principle for data access.

Choosing a protection technique or a combination of various techniques depends on the application and the involved goals. Blanking protects both behavior and identity—only the presence of persons remains perceptible. Therefore, it is usable only in basic surveillance and intruder detection scenarios. Obfuscation and scrambling allow monitoring of the behavior of persons and are therefore more suitable for public safety applications where not only presence of persons but also detection of unusual behavior is of importance. Regardless of the chosen protection technique, two key questions remain the same: (1) is privacy adequately protected by the chosen technique, and (2) what is the impact on the utility of the VSN? Work by Gross et al. [2006] indicates that the overall protection capabilities of pixelation and blurring are relatively low. In more recent work, Dufaux and Ebrahimi [2010] present a framework for the evaluation of privacy protection mechanisms. Their results also show that simple pixelation and blurring offer only limited protection. Blurred or pixelated human faces can often still be identified with standard face recognition algorithms. In contrast to that, the evaluation results indicate that scrambling mechanisms perform much better. A study by Boyle et al. [2000] on the effects of filtered video on awareness and privacy shows that pixelation provides better privacy protection than blurring. Korshunov et al. [2012a] developed an evaluation framework to systematically investigate the privacy protection versus system utility trade-off. The framework consists of a set of standardized questions that are used to assess which information could be observed from a privacy-protected video. For example, questions may cover the gender and race of a person or properties such as worn glasses or scarfs. In this study, a set of video sequences was shown to subjects under controlled lab conditions. The subjects had only a limited amount of time to answer the given questions. The study results indicate that pixelation yields best performance in terms of balance between privacy protection and intelligibility of the video content. Best privacy protection and least intelligibility was achieved with masking filters. Blurring filters resulted in exactly the opposite performance—best intelligibility and least privacy protection. In successive work, Korshunov et al. [2012b] adopted a crowdsourcing approach to get feedback from a larger number of participants. A Facebook application called VideoRate allows users with a valid Facebook account to participate in the evaluation. Results obtained from VideoRate and the previous lab tests are largely consistent.

Abstraction techniques can be tuned to preserve behavior, identity, or both. Finally, encryption is the technique of choice in scenarios where strong but reversible identity protection is required. As with simple blanking, behavior is fully protected and can no longer be monitored. If behavior monitoring is required, a multilevel approach must be chosen.

4.2. Related Work on Privacy

Cavallaro [2004, 2007] emphasizes that digitalization of video surveillance introduces new privacy threats. Therefore, personal and behavioral data should be separated directly on the camera. Whereas system operators only get access to behavioral data, a separate stream containing personal data is made available to law enforcement authorities. Similar ideas are discussed by Senior et al. [2005], who suggest that privacy is protected by extracting and re-rendering sensitive information into multiple, individually protected streams. Fleck and Straßer [2008, 2010] employ smart cameras in an assisted living scenario. The cameras are used to monitor the behavior of persons and detect unusual behavior such as a fall. Detected objects are tracked and their behavior is analyzed using support vector machines. Privacy protection is achieved by either transmitting only event information or replacing detected objects with abstracted versions. It is assumed that the camera's housing is sealed such that manipulation can be detected by the camera and leads to a termination of its services.

Boult [2005] argues that many existing approaches are targeted at removing privacy-sensitive image data without providing mechanisms to reconstruct the original image. Based on this observation, he proposes a system called PICO that relies on cryptography to protect selected image regions. It allows the actions of a person to be monitored without revealing the person's identity. Decryption of faces is performed only in specific circumstances, such as a law violation. Encryption is performed as part of image compression and uses a combination of symmetric and asymmetric cryptography. Additionally, it is suggested that checksums of frames or sub-sequences are computed to ensure data integrity. In related work, Chattopadhyay and Boult [2007] present PrivacyCam, a camera system that identifies regions of interest based on a background model. Resulting regions are encrypted using an AES session key. Rahman et al. [2010] also encrypt regions of interest; they do not rely on established cryptosystems but propose to use chaos cryptography.

Moncrieff et al. [2009] argue that most of the proposed systems rely on predefined security policies and are either too intrusive or too limited. Therefore, they suggest that dynamic data hiding techniques are applied. Via context-based adaptation, the system could remove or abstract privacy-sensitive information during normal operation, whereas in case of an emergency, the full, unmodified video stream is automatically made available. This way, the system remains usable for the intended purpose but protects privacy during normal operation. Dufaux and Ebrahimi [2006] suggest scrambling of sensitive image regions. After detection of relevant areas, images are transformed using DCT. The signs of the coefficients of sensitive regions are then flipped pseudorandomly. The seed for the pseudorandom number generator is encrypted. Decryption is only possible for persons who are in possession of the corresponding decryption key. The main benefits are minimal performance impact and that video streams with scrambled regions can still be viewed with standard players. A similar approach is discussed by Baaziz et al. [2007], where in a first step motion detection is performed followed by content scrambling. To ensure data integrity, an additional watermark is embedded into the image that allows detection of manipulation of image data. Limited reconstruction of manipulated image regions is possible due to redundancy introduced by the watermark. Yabuta et al. [2005] also propose a system where DCT-encoded image data are modified. They, however, do not scramble regions of interest but extract them before DCT encoding and encrypt them. These encrypted regions are then embedded into the DCT-encoded background by modifying the DCT coefficients. Li et al. [2009] present an approach toward recoverable privacy protection based on discreet wavelet transform. Information about sensitive image regions together with their wavelet coefficients are protected with a secret key. Data hiding techniques are used to embed this information into the resulting image.

Qureshi [2009] proposes a framework for privacy protection in video surveillance based on decomposition of raw video into object-video streams. Based on a segmentation approach, pedestrians are identified. Tracking is performed using color features.

The privacy of detected persons is protected by selectively rendering the corresponding objects. The system presented by Tansuriyavong and Hanaki [2001] is also based on detection of sensitive entities. In an office scenario, the silhouettes of detected persons are blanked. Additionally, the system integrates face recognition to identify previously registered persons. Configuration options allow to choose of what information is disclosed—full images, silhouettes, names of known persons, or any combination thereof. Schiff et al. [2007] use visual markers in the form of yellow hardhats to identify persons, and thus sensitive image regions, that are blanked in the outgoing video stream.

Troncoso-Pastoriza et al. [2009] propose a generic video analysis system that is coupled with a DRM system. By exploiting the hierarchical structure of MPEG-4, the authors propose selective visualization of video objects either in clear or in obfuscated forms. Access to sensitive video objects is conditionally granted depending on the rights of the observer and the individual policies of monitored users. Sensitive content is protected by encryption. Intellectual Property Management and Protection (IPMP) descriptors are used to describe the encrypted streams, whereas access rights are formulated using the MPEG-21 Rights Expression Language (REL).

Finally, the Networked Sensor Tapestry (NeST) software architecture by Fidaleo et al. [2004] represents a more generic privacy protection approach. Its design is not limited to videos and images but can handle arbitrary sensor data. The system uses a central component called *privacy buffer*. Data received from the clients is fed into this privacy buffer. The buffer can be extended and configured by means of privacy filters and a privacy grammar. If incoming data are qualified as private by one of the privacy filters, the data do not leave the privacy buffer. Nonprivate data are forwarded to a routing component that manages distribution of data to interested clients.

As summarized in Table II, most privacy protection approaches rely on the identification of sensitive regions. An exception is the work by Saini et al. [2012], which relies on global protection techniques. Blanking is the most common privacy protection technique. Several approaches provide multiple privacy levels where the delivered data stream contains different protection variants of the original sensitive data.

5. USER-CENTRIC SECURITY

People who are monitored by VSNs usually are neither actively asked for consent nor do they have control over their captured personal data. To increase the acceptance of VSNs, data-centric security features such as confidentiality and privacy protection are of utmost importance. As illustrated in Figure 2, nonrepudiation also is a critical aspect for users, as it ensures that they cannot be discredited by manipulated data. But even if these security features are incorporated into the design of a VSN, this is not transparent to users. Therefore, user-centric security must go a step further and provide this transparency in a secure and provable way. Ultimately, an ideal surveillance system should allow users to remain in control over their personal data.

5.1. User-Centric Security Requirements

5.1.1. Awareness and Consent. Monitored people should be made aware of cameras in their environment and their consent should be sought, which can be done via passive or via active methods.

Passive versus Active Methods. Today, simple stickers or signs are used to passively advertise installed camera systems. User consent to monitoring is given implicitly by acknowledging these signs when entering the area. Because the signs are easily overlooked, consent should be sought more actively. Users could be automatically notified about presence and properties of cameras, for example, via their smartphones. If the

Table II. Comparison of Applied Privacy Protection Techniques

	Sensitive Regions	Blanking	Obfuscation / Scrambling	Abstraction	Encryption	Multilevel
Baaziz et al. [2007]	●	○	●	○	○	○
Boult [2005], Chattopadhyay and Boult [2007]	●	○	○	○	●	○
Cavallaro [2004], Cavallaro [2007]	●	●	○	●	○	●
Cheung et al. [2008], Cheung et al. [2006]	●	●	○	○	●	○
Chinomi et al. [2008]	●	●	●	●	○	●
Dufaux and Ebrahimi [2006]	●	○	●	○	●	○
Erdélyi et al. [2013]	●	○	●	◐	○	○
Fidaleo et al. [2004]	●	◐	○	○	○	◐
Fleck and Straßer [2008], Fleck and Straßer [2010]	●	●	●	○	○	○
Korshunov and Ebrahimi [2013]	●	○	●	◐	○	○
Moncrieff et al. [2009]	●	◐	◐	◐	○	●
Qureshi [2009]	●	●	○	●	○	○
Rahman et al. [2010]	●	○	○	○	●	○
Saini et al. [2012]	○	○	◐	●	○	○
Schiff et al. [2007]	●	●	○	○	○	○
Senior et al. [2005]	●	○	○	●	●	●
Tansuriyavong and Hanaki [2001]	●	●	○	●	○	●
Troncoso-Pastoriza et al. [2009]	●	◐	○	○	●	●
Wickramasuriya et al. [2004]	●	●	●	●	○	●
Winkler and Rinner [2011]	●	●	○	●	●	●
Yabuta et al. [2005]	●	●	○	○	●	○

Almost all approaches assume that sensitive image regions are detected and subsequently protected. White bullets represent unsupported, gray bullets partially realized, and black bullets fully covered properties.

system operator discloses data to a third party, explicit user permission should be required.

These requirements have been partially addressed in research prototypes. By handing out dedicated devices or RFID tags to known and trusted users, a stronger form of awareness about video surveillance is realized [Brassil 2005; Wickramasuriya et al. 2004]. Users equipped with such devices are not only made aware of the installed cameras but even get a certain degree of control over their privacy. Cameras recognize them as trustworthy and remove or protect the corresponding image regions. The approach of Cheung et al. [2008] goes even further. By using public key cryptography to protect personal information, users get full control over their privacy-sensitive data because they have to actively participate in the decryption of this data.

Operator versus Crowd-Driven Approaches. Making users aware of installed cameras may not always be in the interest of camera operators. As a consequence, users have taken proactive approaches and started to collect locations of video surveillance cameras on publicly accessible maps on the Internet. Following the spirit of community-driven projects such as Wikipedia, everyone is free to contribute to these databases. One such project is based on OpenStreetMap [OpenStreetMap.org 2011] and makes camera positions available as a map overlay. In February 2010, the city of Paris announced a plan [Prefecture de Police 2010] to establish a police-controlled network of about 1,300 surveillance cameras. Locations of the cameras already installed as well as the planned cameras have been mapped by volunteers on Google Maps [OWNI 2011].

5.1.2. Feedback and Control. In current systems, users have to trust operators to protect their privacy. To establish this trust and give feedback on the internal functionality of the system, Senior et al. [2005] suggest that surveillance equipment should be certified and that the results should be made visible, for example, by stickers attached to cameras. For users, however, it is difficult to evaluate if this certification is still valid. The software of a smart camera might have been changed by the operator without recertification of the system. Therefore, an ideal system should be able to accurately report its current status to users. This report should include information on what personal data are captured, processed, stored, and delivered to observers.

Control goes beyond pure feedback and means to actively involve users whenever their personal data are disclosed to third parties. Asking users for their consent to data disclosure implies that users can be reliably identified and contacted. Identification of captured people is only possible for a system that is used in a controlled environment with a relatively fixed user base such as the employees in a company. For systems deployed in public areas, control is difficult to implement.

5.2. Related Work on User-Centric Security

To protect the privacy of selected users, systems have been presented that allow to remove known, trusted users from captured video. Due to the limited reliability of computer vision algorithms to detect personal image data, many researchers rely on portable devices carried by users for identification and localization. One such approach is presented by Brassil [2005]. He proposes a privacy enabling device (PED) that gives users control over their personal data. When activated, the PED records the location of the person together with timestamps. The data are uploaded to a clearinghouse. Before a camera operator discloses videos to a third party, the clearinghouse has to be contacted to check if an active PED was in the vicinity of the camera at the time in question. If so, video data have to be anonymized.

Wickramasuriya et al. [2004] perform real-time monitoring of the environment to increase user privacy. In particular, they suggest that motion sensors are used to monitor rooms or areas. If motion is detected, an RFID reader is triggered that tries to read the RFID tag carried by the person who entered the area. If no RFID tag can be found or the security level of the tag does not grant access to the area, a camera that oversees the region is activated. Image regions containing persons with valid RFID tags are blanked such that only potential intruders remain visible.

Chinomi et al. [2008] also use RFID technology to detect known users. RFID readers, deployed together with cameras, are used to localize RFID tags carried by users based on signal strength. This location information is then mapped to motion regions detected by the cameras. As the RFID tag identifies the person, the individual privacy policy can be retrieved from a database. This policy defines the relationship between the monitored person and potential observers. Based on that, different forms of abstracted data are delivered by the system. Abstractions include simple dots showing only the location of a person, silhouettes, and blurred motion regions. In addition, Cheung et al. [2008] use RFID for user localization. Corresponding motion regions are extracted from the video and encrypted with the user's public encryption key. This key is retrieved from a database via the user ID from the RFID tag. The blanked regions in the remaining image are filled with background image data using video inpainting [Cheung et al. 2006]. The encrypted regions are embedded into the compressed background image using data hiding techniques similar to steganography. Since decryption of privacy-sensitive image regions requires the user's private key, active user cooperation is necessary to

Table III. Comparison of Implementations of User-Centric Security

	Awareness & Consent	Feedback & Control
Brassil [2005]	●	◐
Cheung et al. [2006], Cheung et al. [2008]	●	◐
Chinomi et al. [2008]	●	○
Schiff et al. [2007]	●	○
Spindler et al. [2006]	◐	○
Wickramasuriya et al. [2004]	●	○
Winkler and Rinner [2012]	●	◐

White bullets represent unsupported, gray bullets partially realized, and black bullets fully covered properties.

reconstruct the original image. In work from the same research group, Ye et al. [2009] and Luo et al. [2010] do not use RFID tags for identification but instead use biometric information. As part of their anonymous biometric access control system, iris scanners are installed at the entrances of areas under video surveillance. Based on that, authorized individuals are then obfuscated in the captured video. Anonymity of authorized persons is maintained by using homomorphic encryption.

An approach that does not need electronic devices that are carried by users is presented by Schiff et al. [2007]. Their “respectful cameras” use visual markers worn by people to identify privacy-sensitive regions. Specifically, they remove person’s faces from images. Spindler et al. [2006] apply similar ideas in the context of building automation and monitoring applications. Personal data are obfuscated based on individual privacy settings. For identification and localization, the authors suggest relying on computer vision. For the prototype, this was not implemented but replaced by manual selection of privacy-sensitive regions.

In our own TrustCAM research we investigated user awareness and trustworthy user feedback [Winkler and Rinner 2010b]. Our visual user-based attestation technique is built on the capabilities of the TPM and its platform status reporting. The underlying idea is that every interested user can query the status of a VSN device and receives a digitally signed report that includes the software which is running on the camera, a corresponding list of system properties as well as information about applied protection and security techniques. In our approach, users employ smartphones to communicate with individual VSN devices. A challenging problem is how to establish a secure communication between the smartphone and the camera. Wireless communication can not be used since it is very difficult to assess if the response is actually coming from the intended camera or from some other, potentially malicious device in the vicinity. Therefore, we developed a protocol [Winkler and Rinner 2012] that relies on visual communication to bootstrap the secure communication channel. Visual communication using 2D barcodes allows users to intuitively select the intended camera and eavesdropping attempts on the communication channel can be easily spotted.

Table III gives an overview of the related work on user-centric security. Awareness and implicit consent of monitored people is achieved in most approaches via special devices (e.g., RFID tags, yellow hard hats) that are carried by protected individuals. Brassil [2005] realizes control over personal data via an intermediate clearinghouse. In addition, Cheung et al. [2006] rely on a mediator to actively involve affected persons upon data disclosure. Feedback about the properties of a surveillance camera and its data protection policies is given in the work of Winkler and Rinner [2012].

6. NODE-CENTRIC SECURITY

Node-centric security subsumes all aspects that relate directly to the security of a VSN device, including both its hardware and software. At first glance, node security might seem less important than the security of the actual data that is captured, processed, and delivered by the VSN node. However, security mechanisms that protect the data are typically situated at the application level. When considering that an attacker might have subverted the node and, for example, has modified the underlying OS or libraries that are used by the applications, then data security is at risk. Once the node has been successfully attacked, it is easy to eavesdrop or modify sensitive data before it is properly protected at the application level. Consequently, node security is a requirement for all high-level data protection techniques.

6.1. Node-Centric Security Requirements

6.1.1. Availability. VSNs often provide important services, and therefore their continuous availability is an important security aspect. Guaranteeing availability is a challenging task, especially when considering that the nodes of a VSN are spatially distributed and that they are not under close physical control of the system operator. Subsequently, we discuss individual VSN availability aspects.

Hardware and Software Denial of Service. Availability of a VSN can be considerably affected by DoS attacks at the hard- or the software level. Typical software DoS attacks try to overwhelm a system by a huge number of incoming service requests sent in a short period of time or by forged requests that consume substantial computing power. The performance of the attacked system is reduced, and legitimate requests can no longer be handled appropriately. Another form of software DoS attack does not target the application level but instead targets the network layer where, for example, routing information is manipulated. Another approach is attacks on the data link layer where a malicious node disrupts the medium access control (MAC) protocols by intentionally causing collisions. An attack similar to DoS is denial of sleep attacks [Raymond et al. 2008] where an attacker exploits MAC functionality to prevent nodes from transitioning into low power states. This way, the small batteries of sensor nodes can be depleted in very short periods of time, such as a few days or even hours.

Hardware attacks on VSNs range from node capturing to the jamming of the radio channel. Physical attacks on VSN availability are even more difficult to prevent than software attacks because of the lack of physical control over the individual nodes.

System Monitoring. A first step toward addressing the availability problem is continuous monitoring of the VSN status. Status monitoring can be implemented via a periodic lifebeat where a VSN node is challenged by a control facility and has to respond in a specific way within a predefined time frame. It must be ensured that lifebeat information is authentic, fresh, and unaltered. In addition to basic availability, lifebeat messages can convey further information on the node's status, including its load and the currently executed applications. Although monitoring does not improve the availability of a VSN, it provides operators with accurate knowledge of the system's status and allows early detection of larger-scale attacks.

Attack Resilience. The services of a VSN should be designed to offer resilience against DoS attacks. Basic protection against illegitimate requests is the authentication of all incoming requests. If sender authentication fails, incoming requests can be discarded immediately. Integrity checks ensure that requests have not been modified by an attacker. To prevent replay attacks, the freshness of incoming requests must be validated either via nonces or timestamps.

Request validation itself can consume a considerable amount of time. When receiving illegitimate requests at a very high rate, a VSN would spend most of its time with checking and discarding those requests. Therefore, request authentication, freshness, and integrity checks are important but insufficient to protect against DoS attacks. To provide reasonable resilience, these techniques have to be complemented with request rate limitation where, for example, intentional and adaptive delays between accepted requests are introduced. Overall, resilience against DoS attacks is a complex topic and service guarantees are hard to achieve, especially in shared, wireless networks.

6.1.2. Physical Security. VSN nodes are typically mounted at easily accessible locations such as walls or poles. Physical attack scenarios range from simple destruction and theft to more sophisticated approaches including hardware manipulation and side channel attacks.

Tamper Detection and Resistance. A VSN device usually comes in a box enclosure that protects the circuit board, sensor, and optics. An attacker might want to get physical access to the circuit to be able to, for example, attach probes to buses to readout on-board memory or to replace individual components. Basic tampering can be prevented when designing the circuit board such that communication lines are not routed at the top or bottom layer of the board. Modern embedded processors allow the memory to be mounted using package on package (POP) techniques where the memory ICs are directly stacked on top of the processor. This packaging technology makes simple memory readout substantially more difficult. Other tamper prevention mechanisms include sealed enclosures or casting ICs and circuit boards with resin. Additional sensors in the node's case or on the circuit board can be used to detect tampering and to take further actions such as automatically erasing or overwriting memory regions.

Side Channels. Side channels attacks exploit characteristics of the circuit such as timing, power consumption [Örs et al. 2004; Popp et al. 2007], or the reaction to intentionally introduced faults. Mitigation of side channel attacks is a difficult task that has to be done as part of IC design.

6.1.3. Code Security. A trend in VSNs and embedded systems in general is that significant portions of the system are implemented in software instead of specialized hardware. While functionality that is implemented in hardware cannot be modified by remote attackers, the software stack of a system is relatively vulnerable.

Authenticity and Integrity. For VSNs, it is important that the software executed on the nodes has been previously approved by the operator or manufacturer and that it has not been modified. Software authentication and integrity checks ensure that an attacker cannot run its own software or that unknown or malicious software is at least detectable. This illustrates the two different approaches in this area: strict enforcement such that only precertified software can be executed versus secure logging and reporting which software is run. To produce meaningful results, both approaches require some minimal support by the hardware platform either as functionality inherent to the CPU or via external hardware such as a TPM.

The enforcement approach is usually called *secure boot*. The SoC provides functionality as part of its boot procedure that allows to check the authenticity and integrity of the executed software (e.g., the bootloader) based on digital signatures and hash sums. If the software cannot be properly validated, it is not executed by the system. For authenticity and integrity checks, a certificate containing the expected hash sum of the verified component must be available. The public signature key corresponding to the certificate has to be available to the SoC for certificate validation and must be well protected against illegitimate modifications. The Mobile Trusted Module (MTM) [Ekberg and Kylänpää 2007] developed by the Trusted Computing Group (TCG) incorporates a secure boot mechanism that is based on similar concepts. For the MTM, there is no strict requirement that it has to be implemented as dedicated hardware. However, in the case of a software MTM, support of the underlying platform such as ARM TrustZone [ARM Limited 2009] or TI M-Shield [Azema and Fayad 2008] is beneficial.

The alternative approach to secure boot is *trusted boot,* which is sometimes also called *authenticated boot*. The major difference to secure boot is that trusted boot securely logs the executed software but does not prevent unknown software from being run. This approach is implemented in the TCG's TPM. The logged device status can be securely reported to an external verifier who can decide if the status of the device is trustworthy or not. Both secure boot and trusted boot only provide information about the status at the time the software is launched. Runtime security issues such as security flaws caused by user input or buffer overruns cannot be captured by these techniques.

Secure Updates and Downgrade Prevention. It is a common practice of manufacturers and operators of VSN devices to install upgraded software either to enhance the functionality of a device or to fix security issues that have been discovered. For this purpose, VSN devices typically come with a remote upgrade mechanism. From a security perspective, it must be ensured that code upgrades can be performed only by known entities (authentication) and that the update was fully received and not modified (integrity). An additional requirement is that the update mechanism only accepts new updates (freshness). Otherwise, an attacker could replay a previous code update and thereby downgrade to a previous software version with exploitable vulnerabilities.

Cloning Prevention. A critical aspect for camera manufacturers is that hardware and software are an inseparable whole. Otherwise, the software of a device could be copied and deployed on a hardware platform with similar features as the original VSN device. For a manufacturer, this is clearly undesirable. To prevent software cloning, the program must be bound to a specific device class or even a unique device. The code itself must be protected against reverse engineering, for example, by means of encryption.

6.2. Related Work on Node-Centric Security

Exhaustively covering all literature that deals with availability, physical security, and code security would be far beyond the scope of this survey. Therefore, we present selected examples that illustrate the state of the art. Availability, DoS attacks, and mitigation approaches have been investigated by various researchers. Common techniques to avoid jamming attacks in wireless networks are frequency hopping spread spectrum (FHSS) and direct sequence spread spectrum (DSSS) [Mpitziopoulos et al. 2009]. In FHSS, the frequency used to transmit data is changed based on a pseudorandom number generator that is initialized with a secret seed that must be known to legitimate communication partners. In DSSS, a sequence of pseudo noise code symbols is used to modulate the transmitted information. Thereby, the original signal is replaced with a very wide bandwidth signal. Again, the pseudo noise sequence has to be known a priori by the transmitter and receiver. In broad- or multicast scenarios where pairwise key setup is not possible, FHSS and DSSS cannot be used directly. Solutions to this problem include uncoordinated frequency hopping and uncoordinated DSSS [Pöpper et al. 2010]. Xu et al. [2006] discuss jamming and argue that FHSS and DSSS are not applicable for typical sensor nodes because their implementation would be too complex and costly. They discuss different jamming detection techniques and outline two principle approaches to achieve resistance against jamming in wireless networks. The first strategy is based on avoiding the jammer in spectral or in physical space by either reallocating the used frequencies or relocating selected nodes in case of a mobile sensor network. The second approach is to compete with the jammer by adjusting transmission power levels or using error correction.

Code security aspects have been investigated by several researchers. Trusted computing and the TPM, even though originally designed for standard PC systems, have found their way onto several WSN and VSN research platforms. If using a standard TPM in an embedded system, an important aspect is that the Low Pin Count (LPC) bus that is used to attach the TPM to a PC is typically not available. Some manufacturers additionally equip their TPMs with a serial, two-wire interface, making them suitable for embedded systems. Grossmann et al. [2007] demonstrate the use of an Atmel AT97SC3203S TPM together with an MSP430 microcontroller from Texas Instruments in the context of a teletherapeutic application. In this scenario, the software state of the embedded device is attested using the TPM before sensitive information is transmitted. secFleck by Hu et al. [2009] is an extension board for the Fleck mote platform [Sikka et al. 2007]. The board is equipped with an Atmel I2C TPM chip. Apparently, the TPM is not used for platform attestation but only for random number generation, for RSA encryption and decryption, and signature creation and verification. In related work, Hu et al. [2010] use the TPM for attesting the status of the sensor node. secFleck is also used by Dua et al. [2009] to enhance security of a participatory sensing application where users sense their local environment and make measurements available to other users. The TPM is used to attest the integrity of the users' platforms.

Aaraj et al. [2008] evaluate the performance of a pure software TPM on an embedded platform (Xscale PXA-250 at 400MHz with 32MB RAM). They present runtime measurements for TPM commands including TPM Quote (1,239ms with a 2,048-bit RSA key) and TPM Sign (902ms, 2,048-bit RSA key). Based on these results, the authors replaced RSA with elliptic curve cryptography (ECC), which reduced the time for TPM Quote to 381ms (224-bit key) and TPM Sign to 191ms (224-bit key). On average, execution time was reduced by a factor of 6.5. ECC is not supported by the current TPM specification but may be adopted in future versions. On another system (Xtensa CPU running at 320Mhz) with partially customizable hardware, the authors implemented dedicated CPU instructions to accelerate ECC. With these hardware optimizations, runtimes for TPM Quote could be reduced to 84.154ms on a unicore and 30.70ms on a hexacore system. Dietrich and Winter [2009, 2010] investigate the possibility of using software-based TPM implementations for embedded systems. Many embedded systems already provide integrated security functionality such as ARM TrustZone that can be used to develop software TPM solutions. The same authors explore the use of smartcards or SIM cards to implement TPM functionality.

Reconfigurable hardware such as FPGA is commonly used in embedded systems. In such a system, not only the software but also the hardware needs to be included in platform attestation. Glas et al. [2008a, 2008b] integrate a TPM with an FPGA system. They introduce a component called Trust-Block that is responsible for securely booting the system. The FPGA itself is split into a user-programmable part and a static section. All reconfiguration of the FPGA has to be performed via functionality provided by this static section. It is also responsible for measuring the new system configuration into the TPM. Eisenbarth et al. [2007] also integrate TC into an FPGA system, but they do not use a dedicated TPM chip; instead, they integrate the TPM functionality as custom logic into the FPGA. Using a so-called Bitstream Trust Engine, they realize authenticity and integrity guarantees. Additionally, they measure the TPM's netlist. The advantage of this approach is that the TPM itself becomes part of the chain of trust; therefore, TPM functionality can be easily updated, extended, and enhanced.

Our TrustCAM prototype is also equipped with a TPM chip [Winkler and Rinner 2010a, 2011]. It is based on an OMAP 3530 ARM SoC from Texas Instruments, has a VGA color CMOS image sensor, and comes with WiFi and ZigBee radios. An Atmel TPM chip is attached via the I2C bus. Among other security features, the system

Table IV. Coverage of Major Node-Centric Security Requirements

	Availability	Physical Security	Code Security
Aaraj et al. [2008]	○	◐	◐
Dietrich and Winter [2009], Winter [2008]	○	◐	●
Dua et al. [2009]	○	◐	●
Glas et al. [2008a], Glas et al. [2008b]	○	◐	◐
Grossmann et al. [2007]	○	◐	●
Hu et al. [2009], Hu et al. [2010]	○	◐	●
Mpitziopoulos et al. [2009]	◐	○	○
Winkler and Rinner [2010a], Winkler and Rinner [2011]	◐	◐	●
Xu et al. [2006]	◐	○	○

White bullets represent unsupported, gray bullets partially realized, and black bullets fully covered properties.

implements trusted boot starting with a TPM-enabled version of the U-Boot bootloader. Measuring the software stack of the system is performed at a coarse granularity where separate measurements of the bootloader stages, the OS kernel, and the root filesystem image are performed. This approach considerably simplifies the effort for recording and reporting the system status. To provide detailed information about the executed applications, the camera middleware takes measurements of the individual computer vision applications that are launched. A trusted lifebeat periodically reports the recorded system status information to a monitoring facility. Additionally, the device status is made available to the user to allow assessment of the security and privacy protection properties of the camera.

Table IV gives an overview of the presented related work on node-centric security. Physical security is addressed usually in limited forms where, for example, tamper resistance for a small portion of the system, such as the storage for cryptographic keys, is realized. Many solutions achieve code security by implementing authenticity and integrity checks for executed software as well as system upgrades. Availability aspects are covered partially via continuous system monitoring or at the radio level via antijamming techniques.

7. NETWORK-CENTRIC SECURITY

One of the four major security domains of Figure 2 is network-centric security, which we partition into channel-related and collaboration-related aspects. Channel security refers to basic protection of the communication channel between two 1:1 communication partners, such as two VSN devices. Collaboration-centric security extends these basic security considerations to networks of VSN nodes that jointly solve given tasks. Typically, VSNs are not directly connected to the Internet but use dedicated basestations for data uplinks. Since they are directly exposed to the Internet, these basestations are the primary target for attackers. Therefore, they need special protection including proactive techniques such as network traffic filtering, firewalls, or the establishment of VPN tunnels between the basestation and a control station. Reactive security techniques such as network intrusion detection systems [Sabahi and Movaghar 2008; Kabiri and Ghorbani 2005] should be considered for these devices to be able to detect unexpected or unusual network traffic and activities.

7.1. Network-Centric Security Requirements

Although many of the security requirements of VSNs overlap with those of WSNs, there are also important differences. A primary one results from the different communication patterns. VSNs typically have to deal with the distribution of large amounts of image and video data within small neighborhoods for the purpose of joint processing. At the same time, low-volume event and control information must be distributed throughout a VSN, relying often on multihop communication. These different traffic types must be also incorporated into the underlying protocols by, for example, adequate prioritization schemes that in turn need to be protected against misuse. Encrypting and digitally signing large amounts of video data is a challenge that is usually not found in WSNs. In this area, VSN platforms benefit from higher on-board computing power required for image analysis as well as from hardware security features found in many novel SoC implementations.

7.1.1. Channel Security. Channel security requires authentication of the communication partners as well as integrity protection, freshness, and confidentiality for transmitted data. These properties are comparable to those achieved via SSL or its successor Transport Layer Security (TLS) [Dierks and Allen 1999]. Data encryption and signing are commonly used techniques to achieve channel security. Real-time encryption of high-volume image and video data can be challenging on low-power VSN systems. With modern SoC devices used for VSN systems [Winkler and Rinner 2013], these issues are alleviated by the overall higher computing power compared to traditional WSN platforms and by dedicated hardware encryption units.

Authenticity, Integrity, Freshness. The requirements for authenticity, integrity, and freshness are similar to those for data-centric security in Section 3. The major difference is that in the context of the network, these requirements apply only for the secure communication channel that is established between two VSN nodes. The security properties are only ensured for the time that the data are in transmission. Once the data arrives at the receiver, the protection no longer applies. Likewise, no guarantees are made for the data before it was transmitted. Protection is only achieved against attacks on the communication link.

Confidentiality. Confidentiality in the context of channel security refers to the protection of transmitted data against eavesdropping by outsiders. Insiders who have access to one of the two communication partners also have full access to the transmitted data in unprotected form assuming that no data-centric security mechanisms have been applied.

7.1.2. Collaboration Security. By collaboration security, we denote network security aspects that go beyond basic 1:1 channel security. This includes MAC and routing protocols, time synchronization, broadcast communication, data sharing, and aggregation, as well as discovery and localization. In many of these protocols, cryptographic keys are required for protection and verification of exchanged information. Whereas many WSN devices are designed around very low-power 8-bit microcontrollers, VSNs devices typically come with more computing power to enable on-board image analysis. This additional computing power of VSN devices can be utilized for asymmetric encryption, which in turn greatly simplifies and strengthens many of the security techniques originally designed for WSNs. Especially session key establishment and data source verification can substantially benefit from these additional capabilities. If asymmetric cryptography is not supported by a specific processing platform, additional chips such as TPMs can be used to add such features.

MAC and Routing. Attacks on the MAC and routing layers have been extensively studied in the context of WSNs [Perrig et al. 2002; Chan and Perrig 2003; Perrig et al. 2004; Wang et al. 2006; Chen et al. 2009; Sen 2010]. The security requirements identified for WSNs also apply to VSNs. MAC layer attacks are primarily targeted at service degradation or interruption. This is achieved via intentionally caused collisions.

In wireless VSNs, low-volume control and event information is often forwarded from source to destination using multihop routing. Large data, such as images or videos, are typically exchanged only in small neighborhoods and very often only via a single or a very small number of hops. Likewise, to keep transmission paths short, high-volume data delivery to monitoring stations is typically handled via dedicated uplink nodes distributed throughout the network. VSN routing protocols must be designed to accommodate these different traffic patterns and to also ensure appropriate prioritization. Without countermeasures in the routing protocol, spoofing or manipulation of routing information can be easily performed by an attacker. By manipulating the routing tables or the route priority information, service quality can be degraded, nodes can be isolated from the network, or substantial amounts of data are directed toward selected nodes, resulting in an overload. If the routing protocol supports integrity protection and authentication, these safeguards can be bypassed if the attacker is in possession of a captured node. Furthermore, the malicious node can be used for selective forwarding where the attacker chooses which information to forward and which not. Other known attacks are, for example, sinkhole attacks where a node makes itself more attractive as a relay by announcing shorter routes. In sybil attacks, a single node simulates the identities of one or more other nodes and collects the data originally intended for these nodes.

Time Synchronization. Every VSN device has its own local clock. For example, to correlate events detected by multiple nodes, a common time base among the participants of the VSN is required. Since the clocks of the VSN nodes operate independently, the time readings of the nodes will differ. These time differences are increased further by the individual drifts of the nodes' oscillators. Consequently, clock and time synchronization is required to enable meaningful comparison of observed events and to jointly solve distributed tasks. Time synchronization mechanisms in wireless networks have been investigated by various researchers [Elson and Römer 2002; Ganeriwal et al. 2003; Sundararaman et al. 2005; Yoon et al. 2007]. From a security perspective, it is apparent that time synchronization protocols are an attractive target for attackers who want to disrupt the services of a VSN. Boukerche and Turgut [2007] distinguish three different groups of attackers on time synchronization. The first group is malicious outsiders who can eavesdrop the communication and who can inject messages. The second group is able to jam to communication channel and can delay and replay captured packages. Finally, the third group includes insiders who have managed to capture a node of the VSN and therefore also have access to the cryptographic keys of the node. Protection against malicious outsiders is based on cryptographic techniques and is not different from protecting any other protocol or data exchange between VSN nodes. Protection against node compromise cannot be achieved solely with cryptographic methods but requires additional node-centric security mechanisms, as discussed in Section 6.

Discovery and Lookup. A key idea of distributed sensor networks is that they operate without fixed or centralized infrastructure. Therefore, decentralized mechanisms are required that allow to discover and query services provided by the members of the network. A typical property is that services can be added or removed at runtime. These characteristics make it difficult to assess which services can be trusted and which are offered by a potentially malicious node. Approaches to identify trustworthy services can be based on service provider authentication using cryptography and unique device

IDs, secure platform status reporting, or reputation concepts where trust decisions are based on the past behavior of a service provider.

Localization and Topology Control. Sensor node locations and topology information are important for aspects such as efficient geographic routing or the avoidance of jammed network regions. The meaning of topology in VSNs is not necessarily identical to that of WSNs. Whereas the focus in WSNs is usually on network topology, the concerns in VSNs are on identifying the nodes' topology with respect to the field of view of the individual cameras. In a tracking scenario, it is important to know which camera most likely will see the object of interest once it leaves the field of view of the current node. This camera network graph [Rowe et al. 2007] can be either predefined by system operators or learned autonomously by the system [Tieu et al. 2005]. This learning phase and potential online topology update mechanisms are interesting for an attacker who wants to manipulate the topology of the VSN such that, for example, actually adjacent cameras are no longer considered in event distribution or that data are disseminated to the wrong cameras.

Broad- and Multicast Communication. In contrast to the previously discussed channel security aspects, collaboration in VSNs typically involves not only 1:1 communication but 1:n communication where, for example, tracking information has to be distributed to all devices within the immediate neighborhood. A primary challenge in this context is the management of cryptographic keys required for message authentication and integrity protection. Asymmetric cryptography is an appropriate tool for authentication and integrity protection in broad- and multicast scenarios. If hardware-accelerated implementations of asymmetric cryptography are available, they can be used to strengthen and simplify the implementation of VSN communication mechanisms. Although such hardware is usually included in state-of-the-art designs, legacy devices might not be powerful enough for asymmetric cryptography. In these situations, symmetric encryption offers far better performance. Keyed hash functions such as HMAC are based on symmetric encryption and shared keys. The fundamental problem in multi- or broadcast scenarios is that the shared key has to be distributed to all members of the group to enable them to authenticate received messages. Being in possession of the shared key enables all members of the group to generate valid messages, which contradicts the idea of individual authentication of messages. Popular approaches that eliminate this shortcoming are the TESLA and uTESLA protocols by Perrig et al. [2002], which are based on hash chaining and delayed disclosure of the symmetric keys.

Data Sharing and Aggregation. A key aspect of VSNs is cooperative scene analysis and object tracking [Micheloni et al. 2005; Velipasalar et al. 2006; Quaritsch et al. 2007; Hoffmann et al. 2008]. For this purpose, object features, meta data, and events have to be shared between individual nodes and joint results are computed based on these aggregated data. Ozdemir and Xiao [2009] discuss the requirements for secure data aggregation and illustrate typical topologies found in data aggregation schemes such as clusters or tree structures. A central question in data aggregation is if and to what extent an aggregator requires access to unprotected data. Ideally, the aggregation process can operate on encrypted data. This is a main difference to data sharing, where communication partners are usually trusted. A critical aspect when sharing data is if the communication partner can guarantee a certain level of security. One approach is to assess the status of the data receiver before data transmission. Remote system status check can, for example, be performed via the attestation capabilities of TPM or MTM chips [Dietrich and Winter 2009; Tan et al. 2010; Kostiainen et al. 2011]. Alternative approaches suggest to make no unprotected data available to communication partners. Secure multiparty computations allow several parties to contribute to a joint result without revealing partial or intermediate results to the participants. Another approach is based on homomorphic encryption [Oleshchuk 2007; Erkin et al. 2009; Hsu et al. 2011], where certain operations can be performed on encrypted data. Although this might be promising for future applications, current solutions are still too limited in their versatility and by far cannot deliver the performance required for real-time computer vision applications.

7.2. Related Work on Network-Centric Security

Karlof and Wagner [2003] examine a wide range of routing protocols that have been proposed for WSNs. None of these protocols have been designed with security in mind, and it comes with no surprise that all studied protocols have severe security flaws. The authors discuss potential mitigation techniques for discovered security problems but conclude that security must be an up-front design goal to realize truly secure routing protocols. INSENS by Deng et al. [2006] is an intrusion-tolerant routing protocol for WSNs. A key design decision for INSENS was that more complex aspects of the protocol are moved away from the resource-constraint devices. A central, resource-rich basestation is the only device that can broad- or multicast data. The basestation is also responsible for creating the forwarding tables for individual nodes. The forwarding tables are based on neighborhood information collected by the nodes and sent subsequently to the basestation. Control and routing information is authenticated using symmetric keys.

The ARAN protocol by Sanzgiri et al. [2005] targets more resource-rich devices and uses asymmetric cryptography. It relies on a certificate server for node authentication that also issues temporal certificates for nodes. The possession of a valid certificate is a requirement for participation in route discovery and internode communication. The Ariadne protocol by Hu et al. [2005] is an on-demand routing protocol based on dynamic source routing. Ariadne makes use of the TESLA broadcast authentication protocol.

Attacks on time synchronization protocols and related security requirements have been investigated by various researchers [Manzo et al. 2005; Boukerche and Turgut 2007]. Secure time synchronization protocols have been proposed, for example, by Ganeriwal et al. [2008], Song et al. [2007], and Sun et al. [2006].

The SPINS security protocol family by Perrig et al. [2002] is one of the most cited security solutions for WSNs. The very limited resources of WSN devices are a primary design aspect of SPINS. SPINS consists of two major parts: the sensor network encryption protocol (SNEP), and uTESLA, which provides secure broadcasting services. SNEP is not limited to data confidentiality but also provides authentication, integrity, and freshness guarantees. For SNEP, the authors assume the availability of a trusted basestation, which is used to establish a shared, symmetric master secret between two nodes. From this shared secret, the communication partners derive three symmetric keys: one is used in the message authentication code, whereas the other two are used for data encryption—one per communication direction. SNEP is complemented by uTESLA, which provides support for secure message broad- and multicast. uTESLA also uses symmetric cryptography together with hash chains. The symmetric keys required by receivers to validate broadcast messages are disclosed by the sender with a delay. Upon disclosure, the sender generates a new key that is used from this point on. A requirement of uTESLA is that the involved nodes have synchronized time sources. Other encryption and authentication protocols designed for WSNs are LEAP+ by Zhu et al. [2006] and TinySEC by Karlof et al. [2004].

Protocols such as WirelessHART, ISA-100.11.a, and ZigBee have been designed for industry and home automation applications. They are used to interact with sensors or to control actuators. Similar to VSNs, these devices are designed for low power consumption, and they typically use multihop communication. Security concepts developed for these types of networks could be adapted and applied for VSN applications.

An ISA-100.11.a [ISA100 Wireless Compliance Institute 2011] network consists of nonrouting sensor and actuator devices as well as routing devices that are responsible for data forwarding but can also incorporate I/O interfaces. Data are transmitted to backbone routers that either route data to other segments of the network or via gateways to higher instances on the network. The ISA-100.11.a stack incorporates various established technologies, including IEEE 802.15.4 as the physical and data link layer, 6LoWPAN as the network layer, or UDP as the transport layer. Confidentiality in ISA-100.11.a is ensured via AES-128 encryption at the data link layer (hop to hop) and in the transport layer (end to end). Data integrity and authenticity are ensured via message integrity codes. The protocol also incorporates protection against replay and delay attacks based on timestamping and nonces. Joining an ISA-100.11.a network involves asymmetric cryptography, whereas the rest of the security functions are based on symmetric cryptography (AES-128).

WirelessHART [HART Communication Foundation 2010] is also used in process automation. Every field device may act as router in a multihop network. A gateway is used as an uplink to higher network segments. WirelessHART is designed as a secure protocol that ensures confidentiality, integrity, authenticity, and freshness of transmitted data [Raza et al. 2009]. Protection can be applied at different levels, providing end-to-end, per-hop, or peer-to-peer security.

In the ZigBee [ZigBee Alliance 2012] protocol, coordinator devices take over the role of a trust center that allows other devices to join the network. The coordinator is also responsible for distribution of cryptographic keys. ZigBee distinguishes three types of keys. Preinstalled master keys are not directly used for encryption but serve as an initial shared secret for key establishment between devices. Network keys are used to protect all messages between nodes within the same ZigBee network. Finally, link keys are used to protect unicast messages between two devices.

Secure aggregation of data has been investigated by various researchers [Wagner 2004; Chan et al. 2006; Westhoff et al. 2006; Castelluccia et al. 2009]. The survey of secure WSN data aggregation schemes by Alzaid et al. [2007] illustrates that most existing schemes are based on symmetric cryptography and message authentication codes. Their studies also show that many aggregation protocols support authentication of involved nodes, as well as confidentiality, integrity, and freshness. A major issue is availability, which is not considered by the examined protocols. To achieve a certain level of availability, the authors suggest to introduce self-healing techniques as well as rotation of the data aggregation nodes.

Table V gives an overview of related work on network-centric security presented in this section. Various researchers have addressed aspects from the field of collaboration security, and a number of protocols have been developed that address security issues at different layers. An open issues is if and how solutions for secure routing, data aggregation, and time synchronization can be combined such that the total overhead is kept at a minimum. Channel security is covered by a set of protocols that have been designed with the requirements and limitations of WSNs and VSNs in mind.

8. VSN SECURITY AND PRIVACY: OBSERVATIONS AND OPEN QUESTIONS

Our analysis of the state of the art of VSN security and privacy protection leads us to six key observations. They summarize the current state and highlight limitations of existing approaches. As a conclusion of the survey, we discuss several directions for future research.

Table V. Coverage of Major Node-Centric Security Requirements

	Channel Security	Collaboration Security
Deng et al. [2006]	○	◐[1]
Castelluccia et al. [2009]	○	◐[2]
Chan et al. [2006]	○	◐[2]
Ganeriwal et al. [2008]	○	◐[3]
Hu et al. [2005]	○	◐[1]
ISA100 Wireless Compliance Institute [2011]	●	○
Karlof et al. [2004]	●	○
Perrig et al. [2002]	●	◐[1]
Sanzgiri et al. [2005]	◐	◐[1]
Song et al. [2007]	○	◐[3]
Sun et al. [2006]	○	◐[3]
Wagner [2004]	○	◐[2]
Westhoff et al. [2006]	○	◐[2]
HART Communication Foundation [2010]	●	○
Zhu et al. [2006]	●	○
ZigBee Alliance [2012]	●	○

[1] MAC and routing.
[2] Data aggregation.
[3] Time synchronization.
White bullets represent unsupported, gray bullets partially realized, and black bullets fully covered properties.

8.1. Key Observations and Limitations

Reactive Data Delivery Cannot Replace Security. Researchers such as Fleck and Straßer [2010] argue that in reactive systems, privacy protection is no longer required because they do not continuously stream data to a monitoring facility as do proactive systems. Rather, reactive systems deliver data only in exceptional situations when a predefined trigger condition has been met. Due to the strong decline of data that is recorded and delivered, reactive systems are much less privacy invasive than proactive systems. However, security issues do not become obsolete. Even if a system delivers only abstracted event descriptions instead of image data, security for abstracted data such as integrity, authenticity, freshness, and confidentiality must be ensured. Furthermore, if a VSN node signals a critical event, an operator might want to request a live video stream to evaluate and assess the reported information before dispatching rescue forces. Whenever a VSN offers the principal possibility of accessing information containing identities or other sensitive personal data, privacy protection is an important issue. From a privacy and security point of view, future VSN designs should become more and more reactive. However, reducing the amount of delivered data does not make security and privacy protection obsolete—which is especially valid for node- and user-centric security aspects.

TradeOff between Privacy Protection and System Utility. The best privacy protection is achieved if a VSN delivers no raw images, no processed images, and no derived data that might reveal the identities of individuals. By delivering no data at all, the highest privacy level is achieved, but at the same time the utility of the VSN vanishes. For the design of secure, privacy-preserving VSNs, it is critical to explore solutions that achieve a reasonable balance between privacy protection and system utility. Due to specific application requirements, different regional laws, and the overall vague notion of privacy, there most likely will not be one single approach but a continuum of solutions.

An aspect that must not be underestimated is privacy loss due to secondary data derived from spatial and temporal correlation of information from multiple cameras, observed behavior and movement patterns of monitored people, and contextual information. So far, these aspects have been barely addressed, and substantial further research is required.

Incomplete Security Requirements and Unclear Responsibilities. Security is rarely considered at design time; if at all, it becomes an issue at later stages of the development process. The actual security requirements depend on the specific usage scenarios for the VSN devices and therefore are either unknown or incomplete during early design phases. A critical aspect is who is responsible for incorporating security and privacy protection and when and at what level they are addressed. Hardware manufacturers leave it to their customers to protect their systems. Firmware and application developers are typically experts in computer vision, machine learning, embedded systems, or related fields. Security is not necessarily their core competence or responsibility. Time-to-market pressure and missing demand for holistic security solutions from operators make security a low-priority topic.

Furthermore, security introduces additional complexity and increases costs for training of developers and operating personnel. On the other hand, security could be a distinguishing feature to set one's products apart from those of competitors. We advocate that security and privacy protection should be turned into off-the-shelf solutions that can be deployed as simply as integrating a different image sensor. Only if security solutions cause very little additional overhead, have been designed by experts, and require only minimal changes to existing designs and established workflows will they then be accepted and adopted by manufacturers, developers, and operators.

Lack of Node-Centric Protection. Most approaches toward VSN security and privacy focus on data security or channel-oriented network security. Security aspects of the VSN platform itself are rarely taken into account. However, without securing the platform, any application-level protection technique can be bypassed if an attacker gets access to a VSN device. Holistic, node-centric security is challenging because VSN devices usually are not under close physical control of operators. Emerging security solutions such as MTM chips or SoC extensions (e.g., ARM TrustZone) are promising concepts for node-centric protection.

Lack of User-Centric Protection. Monitored persons have little knowledge and barely any influence on what data are collected, how the data are used, who has access to the data, or how long the data are stored. VSNs and video surveillance are controversial topics; therefore, it is important not only to integrate adequate security and privacy protection but also to be transparent and open. Only if users are properly notified of installed VSNs and can learn what a system is used for and what data are collected will public acceptance of VSNs be more likely to increase. Researchers are challenged to develop solutions that actively notify users of installed VSNs, get their consent, and allow them to remain in control over their personal data. Obviously, technical solutions alone will be insufficient to address user-centric security and privacy due to the diverging interests of users and system operators. Public demand combined with governmental regulations might be required to foster actual deployment of security and privacy solutions in future VSNs.

Lack of Collaboration Security. In-network processing, data fusion, and cooperative solving of tasks become common in modern VSNs. Security considerations must go beyond protecting the network channel between two VSN devices. Aspects such as secure MAC and routing protocols, as well as time synchronization, have been extensively studied in other research domains such as WSNs. Existing techniques have to be adapted and applied to VSNs. However, there are several challenges in collaboration security that are unique to VSNs. These include secure topology control based on the fields of view of VSN devices or the secure exchange of data among nodes for the purpose of joint image processing. The emerging field of encrypted-domain signal processing is extremely promising for implementation of joint processing. Although cooperating cameras cannot access the encrypted information shared by other cameras, they can still perform signal processing on the provided data. This allows the offloading of computations to adjacent cameras or into the cloud or jointly solving problems as in distributed tracking applications. Encrypted domain processing is enabled by techniques such as homomorphic encryption or secure multiparty computations [Lagendijk et al. 2013; Erkin et al. 2007].

8.2. Open Research Questions

Holistic Security and Privacy Concept. To date, most research on VSN security and privacy focuses on selected, isolated topics. There is a lack of approaches that consider security and privacy in VSNs in a holistic way. Especially apparent in this context is that most solutions are situated at the application level and that node-centric security is not taken into account. A lot of work has been targeted at data- and network-centric security. But without taking the security of VSN devices themselves into account, high-level protection mechanisms are literally built on sand. VSN designers will have to collaborate with engineers from other embedded system domains such as mobile handsets to promote the development of standardized node-centric security solutions. Privacy and security must be seen as primary design goals, and approaches such as privacy by design [Cavoukian 2013a] can help to meet these goals. Another aspect in a holistic security concept is to avoid redundancies. If strong node- and data-centric security are in place, certain network-centric security mechanisms might be obsolete. This, however, cannot be decided at a general basis but depends on the specific application context.

Exploration of VSN Security and Privacy Design Space. For the implementation of security and privacy protection in VSNs, a multidimensional design space exists. The individual dimensions include, for example, required computing power and memory, power consumption, the strength and runtimes of cryptographic algorithms, the level at which protection is applied, and the degree of privacy that is achieved. Privacy protection techniques can be divided into two major groups: object-based protection for sensitive image regions and global approaches that apply uniform protection to the entire image. It is still unclear as to which solutions provide the best protection without substantially reducing system utility.

Another question is how to objectively measure privacy protection. Since privacy depends on personal as well as cultural attitudes, technical approaches alone will be insufficient. A thorough exploration of the privacy protection design space will also have to involve extensive user surveys to determine which privacy protection techniques are appropriate.

Secure and Trustworthy Sensors. So far, most related work has focused on bringing security and privacy protection onto VSN devices. With this approach, one can achieve reasonable protection against attacks on data that is delivered from VSNs to information consumers. However, only limited protection is applied for data while it is on the VSN device. It is an open research topic to identify suitable approaches for on-device data protection. One potential approach is to bring security and privacy protection even closer to the data source by integrating dedicated security functions into the image sensor. If security and privacy are guaranteed at the sensor level, then the camera and its relatively large software stack would no longer have to be implicitly trusted. This raises two major challenges. First, it is unclear what type of privacy protection is suitable and feasible at the sensor level. Second, sensor-level privacy protection means that image processing and analysis applications on the camera must be adapted to deal with preprocessed and prefiltered data.

User Awareness, Feedback, and Control. We have already sketched the need for raising awareness of the presence of VSNs and for giving feedback about the properties, capabilities, and implemented security features of VSNs. As outlined in this survey, early approaches exist that seek to increase user awareness and provide limited user feedback. However, these approaches are still in their infancy. Modern mobile devices such as smartphones open up the possibilities for much more sophisticated approaches where users are proactively notified of VSNs. A central challenge in this context is the development and deployment of a scalable, location- and context-aware notification and feedback system. Even more challenging is user control. The idea is that users who have been recorded by a VSN have to actively give consent whenever their personal data are disclosed to third parties. A fundamental requirement for such a system is the reliable identification of recorded persons to even be able to ask for their consent. Based on the identification of users, personal data could be encrypted with user-specific keys, which ensures that users must be actively involved for data decryption. These concepts are not only technically challenging but also raise a number of ethical questions. Is user identification an appropriate and desirable tool to achieve user control? Wouldn't the inherent requirement for user identification be even worse for user privacy than simple recording of images? Are individuals who can not be identified automatically suspicious? These questions barely scratch the surface and clearly illustrate the need for further, multidisciplinary research.

8.3. Concluding Remarks

VSNs have emerged due to the recent advances in four key disciplines: image sensors, embedded computing, sensor networks, and computer vision. VSNs are expected to become an enabling technology for several applications, and a huge number of deployments are foreseen in public and private places in the near future. Security and privacy protection are crucial properties of these networks, as they capture and process sensitive and private information. In this survey, we captured the state of the art in security and privacy protection in the context of VSNs. Although important contributions have been achieved by the VSN community, a lot of research is still open toward comprehensively secure and privacy-preserving VSNs.

ACKNOWLEDGMENTS

This work was performed as part of the project *TrustEYE: Trustworthy Sensing and Cooperation in Visual Sensor Networks*.[2] The work was supported by funding from the European Regional Development Fund (ERDF) and the Carinthian Economic Promotion Fund (KWF) under grant KWF-3520/23312/35521.

REFERENCES

Najwa Aaraj, Anand Raghunathan, and Niraj K. Jha. 2008. Analysis and design of a hardware/software trusted platform module for embedded systems. *ACM Transactions on Embedded Computing Systems* 8, 1, 1–31.

Hamid Aghajan, Juan Carlos Augusto, Chen Wu, Paul Mccullagh, and Julie-Ann Walkden. 2007. Distributed vision-based accident management for assisted living. In *Proceedings of the International Conference on Smart Homes and Health Telematics*. 196–205.

Kwabene W. Agyeman and Anthony Rowe. 2012. *CMUcam4 Feature List*. Technical Report.

Ian F. Akyildiz, Tommaso Melodia, and Kaushik R. Chowdhury. 2007. A survey on wireless multimedia sensor networks. *Computer Networks* 51, 4, 921–960.

[2] TrustEYE Web site: http://trusteye.aau.at.

M. G. Albanesi, M. Ferretti, and F. Guerrini. 2001. A taxonomy for image authentication techniques and its application to the current state of the art. In *Proceedings of the International Conference on Image Analysis and Processing*. 535–540.

Hani Alzaid, Ernest Foo, and Juan Gonzales Nieto. 2007. Secure data aggregation in wireless sensor network: A survey. In *Proceedings of the Australasian Information Security Conference*. 13.

ARM Limited. 2009. *ARM Security Technology: Building a Secure System Using TrustZone Technology*. Technical Report.

Clemens Arth, Horst Bischof, and Christian Leistner. 2006. TRICam: An embedded platform for remote traffci surveillance. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition Workshop*. 125–133.

Pradeep K. Atrey, Wei-Qi Yan, Ee-Chien Chang, and Mohan S. Kankanhalli. 2004. A hierarchical signature scheme for robust video authentication using secret sharing. In *Proceedings of the International Conference on Multimedia Modelling*. 330–337.

Pradeep K. Atrey, Wei-Qi Yan, and Mohan S. Kankanhalli. 2006. A scalable signature scheme for video authentication. *Multimedia Tools and Applications* 34, 1, 107–135.

Jerome Azema and Gilles Fayad. 2008. *M-Shield Mobile Security Technology: Making Wireless Secure*. Technical Report. Texas Instruments.

Nadia Baaziz, Nathalie Lolo, Oscar Padilla, and Felix Petngang. 2007. Security and privacy protection for automated video surveillance. In *Proceedings of the International Symposium on Signal Processing and Information Technology*. 17–22.

Athanasios Bamis, Dimitrios Lymberopoulos, Thiago Teixeira, and Andreas Savvides. 2010. The BehaviorScope framework for enabling ambient assisted living. *Personal and Ubiquitous Computing* 14, 6 (March 2010), 473–487.

Franco Bartolini, Anastasios Tefas, Mauro Barni, and Ioannis Pitas. 2001. Image authentication techniques for surveillance applications. *Proceedings of the IEEE* 89, 10, 1403–1418.

Horst Bischof, Martin Godec, Christian Leistner, Bernhard Rinner, and Andreas Starzacher. 2010. Autonomous audio-supported learning of visual classifiers for traffic monitoring. *IEEE Intelligent Systems* 25, 3, 15–23.

Azzedine Boukerche and Damla Turgut. 2007. Secure time synchronization protocols for wireless sensor networks. *IEEE Wireless Communications* 14, 5, 64–69.

Terrance Edward Boult. 2005. PICO: Privacy through invertible cryptographic obscuration. In *Proceedings of the Workshop on Computer Vision for Interactive and Intelligent Environments*. 27–38.

Michael Boyle, Christopher Edwards, and Saul Greenberg. 2000. The effects of filtered video on awareness and privacy. In *Proceedings of the Conference on Computer Supported Cooperative Work*. 1–10.

Michael Bramberger, Josef Brunner, Bernhard Rinner, and Helmut Schwabach. 2004. Real-time video analysis on an embedded smart camera for traffic surveillance. In *IEEE Real-Time and Embedded Technology and Applications Symposium*. 174–181.

Jack Brassil. 2005. Using mobile communications to assert privacy from video surveillance. In *Proceedings of the Parallel and Distributed Processing Symposium*. 8.

William M. Bulkeley. 2009. Chicago's Camera Network Is Everywhere. Retrieved March 2012 from http://online.wsj.com/article/SB10001424052748704538404574539910412824756.html.

Claude Castelluccia, Aldar C-F. Chan, Einar Mykletun, and Gene Tsudik. 2009. Efficient and provably secure aggregation of encrypted data in wireless sensor networks. *ACM Transactions on Sensor Networks* 5, 3 (May 2009), 1–36.

Andrea Cavallaro. 2004. Adding privacy constraints to video-based applications. In *Proceedings of the European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology*. 8.

Andrea Cavallaro. 2007. Privacy in video surveillance. *IEEE Signal Processing Magazine* 24, 2 (March 2007), 168–169.

Ann Cavoukian. 2013a. *Privacy by Design Centre of Excellence*. Ph.D. Dissertation. Information & Privacy Commissioner Ontario, Canada.

Ann Cavoukian. 2013b. *Surveillance, Then and Now: Securing Privacy in Public Spaces*. Technical Report.

Haowen Chan and Adrian Perrig. 2003. Security and privacy in sensor networks. *IEEE Computer* 36, 10, 103–105.

Haowen Chan, Adrian Perrig, and Dawn Song. 2006. Secure hierarchical in-network aggregation in sensor networks. In *Proceedings of the International Conference on Computer and Communications Security*. 1–10.

Ankur Chattopadhyay and Terrance Edward Boult. 2007. PrivacyCam: A privacy preserving camera using uCLinux on the Blackfin DSP. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*. 1–8.

Datong Chen, Yi Chang, Rong Yan, and Jie Yang. 2007. Tools for protecting the privacy of specific individuals in video. *EURASIP Journal on Applied Signal Processing* 2007, 1, 107–116.

Mo Chen, Jessica Fridrich, Miroslav Goljan, and Jan Lukás. 2008. Determining image origin and integrity using sensor noise. *IEEE Transactions on Information Forensics and Security* 3, 1, 74–90.

Phoebus Wei-Chih Chen, Parvez Ahammad, Colby Boyer, Shih-I Huang, Leon Lin, Edgar J. Lobaton, Marci Lenore Meingast, Songhwai Oh, Simon Wang, Posu Yan, Allen Yang, Chuohao Yeo, Lung-Chung Chang, Doug Tygar, and S. Shankar Sastry. 2008. CITRIC: A low-bandwidth wireless camera network platform. In *Proceedings of the International Conference on Distributed Smart Cameras*. 10.

Xiangqian Chen, Kia Makki, Kang Yen, and Niki Pissinou. 2009. Sensor network security: A survey. *IEEE Communications Surveys and Tutorials* 11, 2, 52–73.

Sen-Ching Samson Cheung, Jithendra K. Paruchuri, and Thinh P. Nguyen. 2008. Managing privacy data in pervasive camera networks. In *Proceedings of the International Conference on Image Processing*. 1676–1679.

Sen-Ching Samson Cheung, Jian Zhao, and M Vijay Venkatesh. 2006. Efficient object-based video inpainting. In *Proceedings of the International Conference on Image Processing*. 705–708.

Kenta Chinomi, Naoko Nitta, Yoshimichi Ito, and Noboru Babaguchi. 2008. PriSurv: Privacy protected video surveillance system using adaptive visual abstraction. In *Proceedings of the International Multimedia Modeling Conference*. 144–154.

Rita Cucchiara, Massimo Piccardi, and Paola Mello. 2000. Image analysis and rule-based reasoning for a traffic monitoring system. *IEEE Transactions on Intelligent Transportation Systems* 1, 2 (June 2000), 119–130.

Jing Deng, R. Han, and Shivakant Mishra. 2006. INSENS: Intrusion-tolerant routing for wireless sensor networks. *Computer Communications* 29, 2 (January 2006), 216–230.

Tim Dierks and Christopher Allen. 1999. RFC 2246: The TLS Protocol. Retrieved April 2014 from http://www.ietf.org/rfc/rfc2246.txt.

Kurt Dietrich and Johannes Winter. 2009. Implementation aspects of mobile and embedded trusted computing. In *Proceedings of the International Conference on Trust and Trustworthy Computing*. 29–44.

Kurt Dietrich and Johannes Winter. 2010. Towards customizable, application specific mobile trusted modules. In *Proceedings of the Workshop on Scalable Trusted Computing*. 31–40.

Akshay Dua, Nirupama Bulusu, Wu-Chang Feng, and Wen Hu. 2009. Towards trustworthy participatory sensing. In *Proceedings of the USENIX Workshop on Hot Topics in Security*. 6.

Frédéric Dufaux and Touradj Ebrahimi. 2006. Scrambling for video surveillance with privacy. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition Workshop*. 160–166.

Frédéric Dufaux and Touradj Ebrahimi. 2010. A framework for the validation of privacy protection solutions in video surveillance. In *Proceedings of the International Conference on Multimedia and Expo*. 66–71.

Thomas Eisenbarth, Tim Güneysu, Christof Paar, Ahmad-Reza Sadeghi, Dries Schellekens, and Marko Wolf. 2007. Reconfigurable trusted computing in hardware. In *Proceedings of the Workshop on Scalable Trusted Computing*. 15–20.

Jan-Erik Ekberg and Markus Kylänpää. 2007. *Mobile Trusted Module (MTM)—An Introduction*. Technical Report NRC-TR-2007-015. Nokia Research Center.

Jeremy Elson and Kay Römer. 2002. Wireless sensor networks: A new regime for time synchronization. In *Proceedings of the Workshop on Hot Topics in Networks*. 6.

Ádám Erdélyi, Thomas Winkler, and Bernhard Rinner. 2013. Serious fun: Cartooning for privacy protection. In *Proceedings of the MediaEval Workshop*. 2.

Zekeriya Erkin, Martin Franz, Jorge Guajardo, Stefan Katzenbeisser, Inald Lagendijk, and Tomas Toft. 2009. Privacy-preserving face recognition. In *Proceedings of the International Symposium on Privacy Enhancing Technologies*. 235–253.

Zekeriya Erkin, Alessandro Piva, Stefan Katzenbeisser, R. L. Lagendijk, Jamshid Shokrollahi, Gregory Neven, and Mauro Barni. 2007. Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing. *EURASIP Journal on Information Security* 2007, 20.

Dan Farmer and Charles C. Mann. 2003. Surveillance nation (part I). *Technology Review* 4, 34–43.

Douglas A. Fidaleo, Hoang-Anh Nguyen, and Mohan Trivedi. 2004. The Networked Sensor Tapestry (NeST): A privacy enhanced software architecture for interactive analysis of data in video-sensor networks. In *Proceedings of the International Workshop on Video Surveillance and Sensor Networks*. 46–53.

Sven Fleck and Wolfgang Straßer. 2008. Smart camera based monitoring system and its application to assisted living. *Proceedings of the IEEE* 96, 10, 1698–1714.

Sven Fleck and Wolfgang Straßer. 2010. Towards secure and privacy sensitive surveillance. In *Proceedings of the International Conference on Distributed Smart Cameras*. 7.

Gary L. Friedman. 1993. The trustworthy digital camera: Restoring credibility to the photographic image. *IEEE Transactions on Consumer Electronics* 39, 4, 905–910.

Saurabh Ganeriwal, Ram Kumar, and Mani B. Srivastava. 2003. Time-sync protocols for sensor networks. In *Proceedings of the International Conference on Embedded Networked Sensor Systems*. 12.

Saurabh Ganeriwal, Christina Pöepper, Srdjan Capkun, and Mani B. Srivastava. 2008. Secure time synchronization in sensor networks. *ACM Transactions on Sensor Networks* 11, 4, 1–35.

Benjamin Glas, Alexander Klimm, Oliver Sander, Klaus D. Müller-Glaser, and Jürgen Becker. 2008a. A system architecture for reconfigurable trusted platforms. In *Proceedings of the Conference on Design, Automation and Test in Europe*. 541–544.

Benjamin Glas, Alexander Klimm, David Schwab, Klaus D. Müller-Glaser, and Jürgen Becker. 2008b. A prototype of trusted platform functionality on reconfigurable hardware for bitstream updates. In *Proceedings of the International Symposium on Rapid System Prototyping*. 135–141.

Ralph Gross, Latanya Sweeney, Fernando De Torre, and Simon Baker. 2006. Model-based face de-identification. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition Workshop*. 8.

Ulrich Grossmann, Enrik Berkhan, Luciana C. Jatoba, Joerg Ottenbacher, Wilhelm Stork, and Klaus D. Müller-Glaser. 2007. Security for mobile low power nodes in a personal area network by means of trusted platform modules. In *Proceedings of the Workshop on Security and Privacy in Ad Hoc and Sensor Networks*. 172–186.

Arun Hampapur. 2008. Smart video surveillance for proactive security. *IEEE Signal Processing Magazine* 25, 4 (July 2008), 134–136.

Arun Hampapur, S. Borger, Lisa Brown, C. Carlson, Jonathan Connell, Max Lu, Andrew Senior, V. Reddy, Chiao Fe Shu, and Ying-Li Tian. 2007. S3: The IBM smart surveillance system: From transactional systems to observational systems. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*. 1385–1388.

HART Communication Foundation. 2010. WirelessHART Security Overview. Retrieved October 2013 from http://www.hartcomm.org/.

Dajun He, Qibin Sun, and Qi Tian. 2004. A secure and robust object-based video authentication system. *EURASIP Journal on Advances in Signal Processing* 2004, 14, 2185–2200.

Frank Helten and Bernd Fischer. 2004. *What Do People Think about CCTV? Findings from a Berlin Survey*. Technical Report. Berlin Institute for Social Research.

Martin Hoffmann, Michael Wittke, Yvonne Bernard, Ramin Soleymani, and Jörg Hähner. 2008. DMCTRAC: Distributed multi camera tracking. In *Proceedings of the International Conference on Distributed Smart Cameras*. 10.

Chao-Yung Hsu, Chun-Shien Lu, and Soo-Chang Pei. 2011. Homomorphic encryption-based secure SIFT for privacy-preserving feature extraction. In *Proceedings of SPIE*. 12.

Wen Hu, Peter Corke, Wen Chan Shih, and Leslie Overs. 2009. secFleck: A public key technology platform for wireless sensor networks. In *Proceedings of the European Conference on Wireless Sensor Networks*. 296–311.

Wen Hu, Hailun Tan, Peter Corke, Wen Chan Shih, and Sanjay Jha. 2010. Toward trusted wireless sensor networks. *ACM Transactions on Sensor Networks* 7, 1 (August 2010), 25.

Yih-Chun Hu, Adrian Perrig, and David B. Johnson. 2005. Ariadne: A secure on-demand routing protocol for ad hoc networks. *Wireless Networks* 11, 1–2 (January 2005), 21–38.

ISA100 Wireless Compliance Institute. 2011. ISA-100.11.a Wireless Standard. Retrieved October 2013 from http://www.isa100wci.org/.

Peyman Kabiri and Ali A. Ghorbani. 2005. Research on intrusion detection and response: A survey. *International Journal of Network Security* 1, 2, 84–102.

Chris Karlof, Neveen Sastry, and David Wagner. 2004. TinySec: A link layer security architecture for wireless sensor networks. In *Proceedings of the International Conference on Embedded Networked Sensor Systems*. 162–175.

Chris Karlof and David Wagner. 2003. Secure routing in wireless sensor networks: Attacks and countermeasures. *Ad Hoc Networks* 1, 2–3, 293–315.

Vladimir Katalov. 2011. Nikon Image Authentication System: Compromised. Retrieved March 2012 from http://www.elcomsoft.com/nikon.html.

Pavel Korshunov, C. Araimo, Francesca Desimone Simone, C. Velardo, J.-L. Dugelay, and Touradj Ebrahimi. 2012a. Subjective study of privacy filters in video surveillance. In *Proceedings of the International Workshop on Multimedia Signal Processing*. 5.

Pavel Korshunov, Shuting Cai, and Touradj Ebrahimi. 2012b. Crowdsourcing approach for evaluation of privacy filters in video surveillance. In *Proceedings of the International Workshop on Crowdsourcing for Multimedia*. 6.

Pavel Korshunov and Touradj Ebrahimi. 2013. Using warping for privacy protection in video surveillance. In *Proceedings of the International Conference on Digital Signal Processing*. 6.

Kari Kostiainen, N. Asokan, and Jan-Erik Ekberg. 2011. Practical property-based attestation on mobile devices. In *Proceedings of the International Conference on Trust and Trustworthy Computing*. 78–92.

Stefan Krempl and Andreas Wilkens. 2011. Datenschützer beanstanden Videoüberwachung in ECE-Einkaufszentren.Retrieved March 2013 from http://heise.de/-1187205.

Reginald L. Lagendijk, Zekeriya Erkin, and Mauro Barni. 2013. Encrypted signal processing for privacy protection. *IEEE Signal Processing Magazine* 30, 1, 82–105.

Chang-tsun Li. 2010. Source camera identification using enhanced sensor pattern noise. *IEEE Transactions on Information Forensics and Security* 5, 2 (June 2010), 280–287.

Guangzhen Li, Yoshimichi Ito, Xiaoyi Yu, Naoko Nitta, and Noboru Babaguchi. 2009. Recoverable privacy protection for video content distribution. *EURASIP Journal on Information Security* 2009, 11.

Ying Luo, Shuiming Ye, and Sen-Ching Samson Cheung. 2010. Anonymous subject identification in privacy-aware video surveillance. In *Proceedings of the International Conference on Multimedia and Expo*. 83–88.

Michael Manzo, Tanya Roosta, and Shankar Sastry. 2005. Time synchronization attacks in sensor networks. In *Proceedings of the Workshop on Security of Ad Hoc and Sensor Networks*. 10.

Andrew Martin. 2008. *The Ten Page Introduction to Trusted Computing*. Technical Report RR-08-11. Oxford University Computing Laboratory.

Isabel Martínez-Ponte, Xavier Desurmont, Jerome Meessen, and Jean-François Delaigle. 2005. Robust human face hiding ensuring privacy. In *Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services*. 4.

Nasir Memon and Ping Wah Wong. 1998. Protecting digital media content. *Communications of the ACM* 41, 7 (July 1998), 35–43.

Christian Micheloni, Gian Luca Foresti, and Lauro Snidaro. 2005. A network of co-operative cameras for visual surveillance. *IEEE Proceedings on Vision, Image and Signal Processing* 152, 2, 205–212.

Saraju P. Mohanty. 2009. A secure digital camera architecture for integrated real-time digital rights management. *Journal of Systems Architecture* 55, 10–12 (October 2009), 468–480.

Simon Moncrieff, Svetha Venkatesh, and Geoff West. 2009. Dynamic privacy in public surveillance. *IEEE Computer* 42, 9 (September 2009), 22–28.

Aristides Mpitziopoulos, Damianos Gavalas, Charalampos Konstantopoulos, and Grammati Pantziou. 2009. A survey on jamming attacks and countermeasures in WSNs. *IEEE Communications Surveys and Tutorials* 11, 4, 1–14.

Steven Ney and Kurt Pichler. 2002. *Video Surveillance in Austria*. Technical Report. Interdisciplinary Centre for Comparative Research in the Social Sciences, Austria.

Clive Norris. 2009. *A Review of the Increased Use of CCTV and Video-Surveillance for Crime Prevention Purposes in Europe*. Technical Report. Department of Sociological Studies, University of Sheffield, United Kingdom.

Vladimir A. Oleshchuk. 2007. Privacy preserving monitoring and surveillance in sensor networks. In *Proceedings of the Frontiers of High Performance Computing and Networking Workshops.* Lecture Notes in Computer Science, Vol. 4743. 485–492.

OpenStreetMap.org. 2011. OpenStreetMap.org Video Surveillance Camera Overlay. Retrieved April 2011 from http://osm.vdska.de/.

Sddka Berna Örs, Gürkaynak Frank, Elisabeth Oswald, and Bart Preneel. 2004. Power-analysis attack on an ASIC AES implementation. In *Proceedings of the International Conference on Information Technology: Coding and Computing*. 2–8.

OWNI. 2011. CCTV Camera Positions in Paris. Retrieved March 2012 from http://app.owni.fr/camera-paris/.

Suat Ozdemir and Yang Xiao. 2009. Secure data aggregation in wireless sensor networks: A comprehensive overview. *Computer Networks* 53, 12 (August 2009), 2022–2037.

Adrian Perrig, Ran Canetti, J. D. Tygar, and Dawn Song. 2002. The TESLA broadcast authentication protocol. *RSA Cryptobytes* 5.

Adrian Perrig, John A. Stankovic, and David Wagner. 2004. Security in wireless sensor networks. *Communications of the ACM* 47, 6, 53–57.

Adrian Perrig, Robert Szewczyk, J. D. Tygar, Victor Wen, and David E. Culler. 2002. SPINS: Security protocols for sensor networks. *Wireless Networks* 8, 5, 521–534.

Alvaro Pinto. 2011. *Wireless Embedded Smart Cameras: Performance Analysis and Their Application to Fall Detection for Eldercare*. Ph.D. Dissertation.

Thomas Popp, Stefan Mangard, and Elisabeth Oswald. 2007. Power analysis attacks and countermeasures. *IEEE Design and Test of Computers* 24, 6, 535–543.

Christina Pöpper, Mario Strasser, and Srdjan Capkun. 2010. Anti-jamming broadcast communication using uncoordinated spread spectrum techniques. *IEEE Journal on Selected Areas in Communications* 28, 5, 1–13.

Prefecture de Police. 2010. *Plan de Vidéoprotection pour Paris*. Technical Report.

Markus Quaritsch, Markus Kreuzthaler, Bernhard Rinner, Horst Bischof, and Bernhard Strobl. 2007. Autonomous multicamera tracking on embedded smart cameras. *EURASIP Journal on Embedded Systems* 2007, 1, 10.

Jean-Jacques Quisquater, Benoit Macq, Marc Joye, N. Degand, and A. Bernard. 1997. Practical solution to authentication of images with a secure camera. *Storage and Retrieval for Image and Video Databases* 3022, 1, 290–297.

Faisal Z. Qureshi. 2009. Object-video streams for preserving privacy in video surveillance. In *Proceedings of the International Conference on Advanced Video and Signal-Based Surveillance*. 442–447.

Mohammad Rahimi, Rick Baer, Obimdinachi I. Iroezi, Juan C. Garcia, Jay Warrior, Deborah Estrin, and Mani B. Srivastava. 2005. Cyclops: In situ image sensing and interpretation in wireless sensor networks. In *Proceedings of the International Conference on Embedded Networked Sensor Systems*. 13.

Mizanur Rahman, M. Anwar Hossain, Hussein Mouftah, A. El Saddik, and Eiji Okamoto. 2010. A real-time privacy-sensitive data hiding approach based on chaos cryptography. In *Proceedings of the International Conference on Multimedia and Expo*. 72–77.

David R. Raymond, Scott F. Midkiff, Anthony Wood, and John A. Stankovic. 2008. Denial-of-service in wireless sensor networks: Attacks and defenses. *IEEE Pervasive Computing* 7, 1, 74–81.

Shahid Raza, Adriaan Slabbert, Thiemo Voigt, and Krister Landernas. 2009. Security considerations for the WirelessHART protocol. In *Proceedings of the International Conference on Emerging Technologies and Factory Automation*. 8.

Bernhard Rinner and Wayne Wolf. 2008. A bright future for distributed smart cameras. *Proceedings of the IEEE* 96, 10 (October 2008), 1562–1564.

Anthony Rowe, Dhiraj Goel, and Raj Rajkumar. 2007. FireFly mosaic: A vision-enabled wireless sensor networking system. In *Proceedings of the International Real-Time Systems Symposium*. 459–468.

F. Sabahi and A. Movaghar. 2008. Intrusion detection: A survey. In *Proceedings of the International Conference on Systems and Networks Communications*. 23–26.

Mukesh Saini, Pradeep K. Atrey, Sharad Mehrotra, Sabu Emmanuel, and Mohan S. Kankanhalli. 2010. Privacy modeling for video data publication. In *Proceedings of the International Conference on Multimedia and Expo*. 60–65.

Mukesh Saini, Pradeep K. Atrey, Sharad Mehrotra, and Mohan S. Kankanhalli. 2012. Adaptive transformation for robust privacy protection in video surveillance. *Advances in Multimedia*, 1–10.

Kimaya Sanzgiri, Daniel Laflamme, Bridget Dahill, Brian Neil Levine, Clay Shields, and Elizabeth M. Belding-Royer. 2005. Authenticated routing for ad hoc networks. *IEEE Journal on Selected Areas in Communications* 23, 3, 598–610.

Martin Schaffer and Peter Schartner. 2007. Video surveillance: A distributed approach to protect privacy. In *Proceedings of the International Conference on Communications and Multimedia Security*. 140–149.

Jeremy Schiff, Marci Meingast, Deirdre K. Mulligan, Shankar Sastry, and Kenneth Y. Goldberg. 2007. Respectful cameras: Selecting visual markers in real-time to address privacy concerns. In *Proceedings of the International Conference on Intelligent Robots and Systems*. 971–978.

Adolph Seema and Martin Reisslein. 2011. Towards efficient wireless video sensor networks: A survey of existing node architectures and proposal for a Flexi-WVSNP design. *IEEE Communications Surveys and Tutorials* 13, 3, 462–486.

Jaydip Sen. 2010. A survey on wireless sensor network security. *International Journal of Communication Networks and Information Security* 1, 2 (November 2010), 55–78.

Andrew Senior, Sharath Pankanti, Arun Hampapur, Lisa Brown, Ying-Li Tian, Ahmet Ekin, Jonathan Connell, Chiao Fe Shu, and Max Lu. 2005. Enabling video privacy through computer vision. *IEEE Security and Privacy Magazine* 3, 3, 50–57.

Dimitrios N. Serpanos and Andreas Papalambrou. 2008. Security and privacy in distributed smart cameras. *Proceedings of the IEEE* 96, 10 (October 2008), 1678–1687.

Stefaan Seys, Claudia Diaz, Bart De Win, Vincent Naessens, Caroline Goemans, Joris Claessens, Wim Moreau, Bart De Decker, Jos Dumortier, and Bart Preneel. 2001. *APES Anonymity and Privacy in Electronic Services*. Technical Report.

Pavan Sikka, Peter Corke, Leslie Overs, Philip Valencia, and Tim Wark. 2007. Fleck: A platform for real-world outdoor sensor networks. In *Proceedings of the International Conference on Intelligent Sensors, Sensor Networks and Information*. 709–714.

Dmitry Sklyarov. 2010. Forging Canon Original Decision Data. Retrieved April 2011 from http://www.elcomsoft.com/canon.html.

Hui Song, Sencun Zhu, and Guohong Cao. 2007. Attack-resilient time synchronization for wireless sensor networks. *Ad Hoc Networks* 5, 1 (January 2007), 112–125.

Stanislava Soro and Wendi B. Heinzelman. 2009. A survey of visual sensor networks. *Advances in Multimedia* 2009 (May 2009), 21.

Torsten Spindler, Christoph Wartmann, Ludger Hovestadt, Daniel Roth, Luc van Gool, and Andreas Steffen. 2006. Privacy in video surveilled areas. In *Proceedings of the International Conference on Privacy, Security and Trust*. 10.

Kun Sun, Peng Ning, Cliff Wang, An Liu, and Yuzheng Zhou. 2006. TinySeRSync: Secure and resilient time synchronization in wireless sensor networks. In *Proceedings of the Conference on Computer and Communications Security*. 14.

Bharath Sundararaman, Ugo Buy, and Ajay D. Kshemkalyani. 2005. Clock synchronization for wireless sensor networks: A survey. *Ad Hoc Networks* 3, 4, 281–323.

Y. Sutcu, S. Bayram, H. T. Sencar, and Nasir Memon. 2007. Improvements on sensor noise based source camera identification. In *Proceedings of the International Conference on Multimedia and Expo*. 24–27.

Hailun Tan, Wan Hu, and Sanjay Jha. 2010. A hardware-based remote attestation protocol in wireless sensor networks. In *Proceedings of the International Conference on Information Processing in Sensor Networks*. 378–379.

Suriyon Tansuriyavong and Shinichi Hanaki. 2001. Privacy protection by concealing persons in circumstantial video image. In *Proceedings of the Workshop on Perceptive User Interfaces*. 4.

Kinh Tieu, Gerald Dalley, and W. Eric L. Grimson. 2005. Inference of non-overlapping camera network topology by measuring statistical dependence. In *Proceedings of the International Conference on Computer Vision*. 1842–1849.

Juan R. Troncoso-Pastoriza, Luis Pérez-Freire, and Fernando Pérez-González. 2009. Videosurveillance and privacy: Covering the two sides of the mirror with DRM. In *Proceedings of the Workshop on Digital Rights Management*. 83–94.

Khai N. Truong, Shwetak N. Patel, Jay W. Summet, and Gregory D. Abowd. 2005. Preventing camera recording by designing a capture-resistant environment. In *Proceedings of the International Conference on Ubiquitous Computing*. 73–86.

Trusted Computing Group. 2011. TPM Main Specification 1.2, Level 2, Revision 116. Retrieved October 2013 from http://www.trustedcomputinggroup.org/resources/tpm_main_specification.

Hauke Vagts and Alexander Bauer. 2010. Privacy-aware object representation for surveillance systems. In *Proceedings of the International Conference on Advanced Video and Signal-Based Surveillance*. 601–608.

Hauke Vagts and Jürgen Beyerer. 2009. Security and privacy challenges in modern surveillance systems. In *Proceedings of the Future Security Research Conference*. 94–116.

Senem Velipasalar, Jason Schlessman, Cheng-yao Chen, Wayne Wolf, and Jaswinder Pal Singh. 2006. SCCS: A scalable clustered camera system for multiple object tracking communicating via message passing interface. In *Proceedings of the International Conference on Multimedia and Expo*. 277–280.

David Wagner. 2004. Resilient aggregation in sensor networks. In *Proceedings of the Workshop on Security of Ad Hoc and Sensor Networks*. 10.

Yong Wang, Garhan Attebury, and Byrav Ramamurthy. 2006. A survey of security issues in wireless sensor networks. *IEEE Communications Surveys and Tutorials* 8, 2, 2–23.

Dirk Westhoff, Joao Girao, and Mithun Acharya. 2006. Concealed data aggregation for reverse multicast traffic in sensor networks: Encryption, key distribution, and routing adaptation. *IEEE Transactions on Mobile Computing* 5, 10, 1417–1431.

Jehan Wickramasuriya, Mahesh Datt, Sharad Mehrotra, and Nalini Venkatasubramanian. 2004. Privacy protecting data collection in media spaces. In *Proceedings of the International Conference on Multimedia*. 48–55.

Thomas Winkler and Bernhard Rinner. 2010a. TrustCAM: Security and privacy-protection for an embedded smart camera based on trusted computing. In *Proceedings of the International Conference on Advanced Video and Signal-Based Surveillance*. 593–600.

Thomas Winkler and Bernhard Rinner. 2010b. User-based attestation for trustworthy visual sensor networks. In *Proceedings of the International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*. 8.

Thomas Winkler and Bernhard Rinner. 2011. Securing embedded smart cameras with trusted computing. *EURASIP Journal on Wireless Communications and Networking* 2011, 20.

Thomas Winkler and Bernhard Rinner. 2012. User centric privacy awareness in video surveillance. *Multimedia Systems Journal* 18, 2, 99–121.

Thomas Winkler and Bernhard Rinner. 2013. Sensor-level security and privacy protection by embedding video content analysis. In *Proceedings of the International Conference on Digital Signal Processing*. 6.

Johannes Winter. 2008. Trusted computing building blocks for embedded Linux-based arm trustzone platforms. In *Proceedings of the Workshop on Scalable Trusted Computing*. 21–30.

Wayne Wolf, Burak Ozer, and Tiehan Lv. 2003. Architectures for distributed smart cameras. In *Proceedings of the International Conference on Multimedia and Expo*. 5–8.

Wenyuan Xu, Ke Ma, Wade Trappe, and Yanyong Zhang. 2006. Jamming sensor networks: Attack and defense strategies. *IEEE Network* 20, 3, 41–47.

Kenichi Yabuta, Hitoshi Kitazawa, and Toshihisa Tanaka. 2005. A new concept of security camera monitoring with privacy protection by masking moving objects. In *Proceedings of the International Pacific-Rim Conference on Multimedia*. 831–842.

Shuiming Ye, Ying Luo, Jian Zhao, and Sen-Ching Samson Cheung. 2009. Anonymous biometric access control. *EURASIP Journal on Information Security* 2009, 18.

Suyoung Yoon, Chanchai Veerarittiphan, and Mihail L. Sichitiu. 2007. Tiny-sync: Tight time synchronization for wireless sensor networks. *ACM Transactions on Sensor Networks* 3, 2, 1–33.

Yun Zhou, Yuguang Fang, and Yanchao Zhang. 2008. Securing wireless sensor networks: A survey. *IEEE Communications Surveys and Tutorials* 10, 3, 6–28.

Sencun Zhu, Sanjeev Setia, and Sushil Jajodia. 2006. LEAP+: Efficient security mechanisms for large-scale distributed sensor networks. *ACM Transactions on Sensor Networks* 2, 4, 500–528.

ZigBee Alliance. 2012. ZigBee Consortium Web Site. Retrieved October 2013 from http://www.zigbee.org/.

Received January 2013; revised July 2013; accepted November 2013