Privacy-protecting Smart Cameras

Bernhard Rinner and Thomas Winkler
Institute of Networked and Embedded Systems and Lakeside Labs
Alpen-Adria-Universität Klagenfurt
Lakeside Park B02b, 9020 Klagenfurt
bernhard.rinner@aau.at, thomas.winkler@aau.at

ABSTRACT

Smart cameras are considered as emerging technology for the realization of various smart environments ranging from room to city scales. Since these cameras capture images that potentially reveal sensitive information about individuals, appropriate protection mechanisms are required. In this paper we discuss the key security and privacy protection domains in smart camera networks and present our TrustEYE.M4 platform which embeds privacy protection close to the image sensor and provides strong separation between hardware-supported protection and flexible scene analytics.

Categories and Subject Descriptors

K.6.5 [**Management of computing and information systems**]: Security and Protection—*Authentication, Unauthorized access*; I.4.9 [**Image processing and computer vision**]: Applications; C.3 [**Computer Systems Organization**]: Special-purpose and application-based systems—*Real-time and embedded systems*

General Terms

Security, Privacy

Keywords

Visual sensors, Embedded smart cameras, Security, Privacy

1. INTRODUCTION AND MOTIVATION

Smart camera networks are real-time, distributed, embedded systems that perform computer vision tasks using multiple cameras [23, 24, 1]. They are considered an emerging technology for various applications including surveillance, entertainment and smart environments. Although there exists a wide variety in applications a common issue is that these cameras capture images that potentially reveal sensitive information about individuals such as their identities or interaction patterns. Thus, privacy protection is an important requirement for such applications.

While privacy protection is not a novel topic, it gains importance in smart camera networks for several reasons. First, captured visual information can be easily interpreted by non-experts making it an attractive target for misuse. Second, privacy protection in camera networks has mostly not been addressed in a holistic approach. Finally, recent trends towards open networks and architectures including cloud-based services require advanced protection mechanisms.

The objectives of this paper are twofold. On the one hand, we briefly identify the key domains of current research of security and privacy protection in smart camera networks. On the other hand, we present our TrustEYE.M4 platform which embeds privacy protection close to the image sensor and provides a clear separation between hardware-supported protection and flexible scene analytics. Such an approach is advantageous in many smart environment applications.

2. SMART CAMERA SECURITY

Privacy protection in video-centric applications needs to be implemented as early in the processing pipeline as possible—making privacy protection an integral part of the sensing device is therefore crucial. As a consequence, the security of the camera device including both its hard- and software and secure network communication become critical aspects. We subsequently present a holistic classification approach that divides smart camera security and privacy protection into four domains which are shown in Figure 1. Data-centric security addresses security and privacy aspects of captured raw images and all derived high-level information. Node-centric security subsumes all aspects related to the camera platform while network-centric security extends these considerations to inter-camera communication. Finally, user-centric security deals with security and privacy aspects related to people monitored by the camera system. We subsequently discuss these four domains in more detail.

Data-centric security. It addresses the protection of all data that is made available by a camera system. The definition of data in this context is not limited to raw images but includes also processed image data, all types of derived information as well as high-level event descriptions. For all delivered data *non-repudiation* as well as *confidentiality* must be ensured [28]. In our classification, data-centric security properties are tightly bound to the data and have the same lifetime as the data.

Non-repudiation and confidentiality are well defined goals

ICDSC '14, November 04 - 07 2014, Venezia Mestre, Italy.
Copyright 2014 ACM 978-1-4503-2925-5/14/11...$15.00.
http://dx.doi.org/10.1145/2659021.2659044

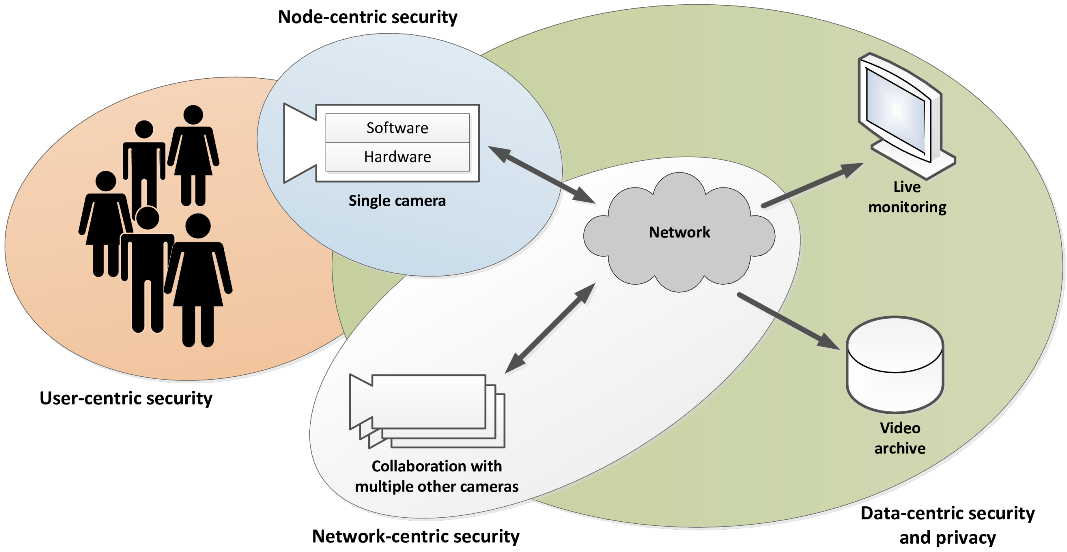

Figure 1: Smart camera security is broken down into four domains. Data-centric security focuses on non-repudiation and confidentiality for recorded data for its entire lifetime. Node-centric security refers to all aspects directly related to the embedded camera device including both its soft- and its hardware. Network-centric security addresses 1:1 and 1:n communication. User-centric security means making monitored people aware of cameras and giving them the possibility to check if and how their personal data is protected.

for data security. Non-repudiation includes authenticity of images and videos (i.e., which camera captured the data), when it was recorded and where. Typical approaches to ensure non-repudiation are digital signatures [2, 34] or watermarks [18], secure timestamping mechanisms and possibly localization techniques such as GPS. Confidentiality denotes the protection of images, videos as well as all derived data against access by external parties [27]. Confidentiality must be maintained throughout the entire lifetime of the data starting from image capturing and going to long-term archiving in a database. It is typically achieved via data encryption. Internal parties such as system operators or security guards require access to confidential information to fulfill their duties.

Privacy is a sub-property of confidentiality which denotes protection of sensitive data against misuse by legitimate users (i.e., insiders such as security guards). For system operators who perform monitoring tasks, behavioral information is usually sufficient and identity information is not required [6, 7]. This can be achieved by automatic detection and removal of sensitive image regions such as people's faces [9, 22].

Node-centric security. It subsumes all aspects that relate directly to the security of a smart camera device including both its hard- and its software. At first glance, node security might seem less important than the security of the actual data that is captured, processed and delivered by a smart camera. 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, e.g., 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. Node-centric security aspects include physical platform security, code security and secure system monitoring [34], availability and resistance against denial of service attacks.

Network-centric security. We partition network-centric security 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 camera devices. Collaboration-centric security extends these basic security considerations to networks of smart cameras which jointly solve given tasks.

The requirements for non-repudiation and confidentiality are similar to those for data-centric security. The major difference is that in the context of the network these requirements apply only for the secure communication channel that is established between nodes. The security properties are only ensured for the time the data is 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. These properties are realized, e.g., by SSL or its successor Transport Layer Security (TLS) [11]. Collaboration security denotes network security aspects which go beyond basic channel security. This includes secure MAC and routing protocols [14, 26, 13], secure time synchronization [15, 3], broadcast communication [20], data sharing and aggregation [19, 29, 8, 30, 5], as well as discovery and localization.

User-centric security. By user-centric security we address people who are monitored by smart cameras; they usually are neither actively asked for consent nor do they have control over their captured personal data. To increase the acceptance of monitoring systems, data-centric security features such as confidentiality and privacy protection are of utmost importance. But even if these security features are incorporated into the design of a smart camera, this is not transparent for users. Therefore, user-centric security must go a step further and provide this transparency in a secure an provable way. Part of this effort is to make users aware of the cameras in their environment, to actively seek user consent and to give feedback what data is captured, for what purpose, by whom and how long it is stored. Ultimately, an ideal smart environment should allow users to remain in control over their personal data. Approaches in this direction augment camera systems with additional technologies such as RFID for user identification. By handing out dedicated devices or RFID tags to known and trusted users, a stronger form of awareness about video surveillance is realized [4, 31]. 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. This approach is taken a step further by using public key cryptography to protect personal information [10]. Users get full control over their privacy-sensitive data since they have to actively participate in the decryption of this data.

3. TRUSTEYE.M4 PLATFORM

The main motivation for our secure sensing unit approach is to perform security and privacy protection as close as possible to the image sensor. As shown in Figure 2, the camera device is divided into a secure sensing unit and a camera host system. The secure sensing unit has exclusive access to the raw image data and applies data security techniques and image pre-filtering for privacy protection. Data security typically provides non-repudiation guarantees [18, 35] (i.e., authenticity, integrity and freshness/timestamping) for captured images. This can be achieved via cryptographic techniques [34] or via image watermarking [17] and steganography. Privacy protection is implemented by filtering captured images before they are forwarded to the camera host system. Filtering can be performed for regions of interest (object-based) or for the entire image (global). Object-based filtering [21] typically targets human bodies or faces [16]. The achieved privacy protection depends on the performance of the underlying object detection algorithm. Global techniques [12, 25] filter the entire frame and therefore do not depend on the detection performance. Adaptive global protection combines both approaches by incorporating the results of unreliable detectors [25] to determine the strength of global protection. We have developed TrustEYE.M4—a custom-designed hard- and software platform. The main design goal is its application as a secure sensing unit together with an off-the-shelf camera host system. For that application, hardware components have been selected which provide a dedicated high-performance image sensor interface and hardware security features for firmware protection.

The TrustEYE.M4 sensing unit shown as top module in Figure 3 is based on a two layer 50×50 mm printed circuit board and is using an STM32F417 ARM Cortex M4 microcontroller. The CPU provides 192 kB on-chip SRAM and 1 MB on-chip program Flash memory. Since the onchip SRAM is insufficient to hold multiple images for processing and since typical computer vision algorithms require additional storage for intermediate results, an additional 2×2 MB of external SRAM are included on the circuit board. Data transfers from the image sensor module to SRAM and from SRAM to the camera host system are implemented via the microcontroller's DMA engines such that the CPU itself is available for image processing. The system is powered either via a Micro-USB connector, a single-cell lithium polymer battery or directly via the camera host system. Currently two image sensor modules are supported—one with an OmniVision OV7725 (640×480) and one with an OmniVision OV5642 sensor (5 megapixels). The sensors are configured via the I2C bus, deliver their data via a parallel 8-bit interface and can be configured for various data formats including YUV422, YUV420 or RGB. Programming and debugging support is provided via the Serial Wire Debug (SWD) interface or the controller's serial bootloader. An dedicated connector (cp. Figure 3) attaches TrustEYE.M4

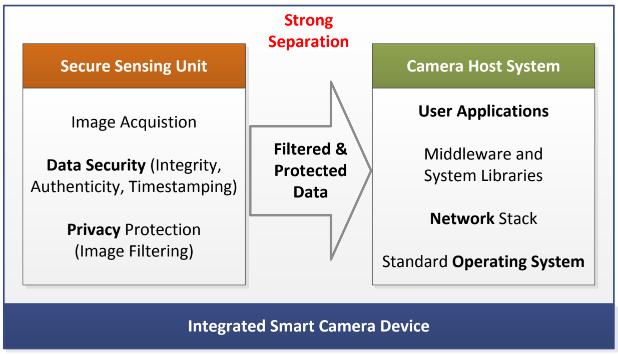
Strong
Separation
Secure Sensing Unit
Image Acquistion
Data Security (Integrity, Authenticity, Timestamping)
Privacy Protection (Image Filtering)
Filtered & Protected Data
Camera Host System
User Applications
Middleware and System Libraries
Network Stack
Standard Operating System
Integrated Smart Camera Device

Figure 2: The camera is divided into a secure sensing unit with exclusive access to raw images and an untrusted camera host system which runs user applications, middleware and networking tasks.

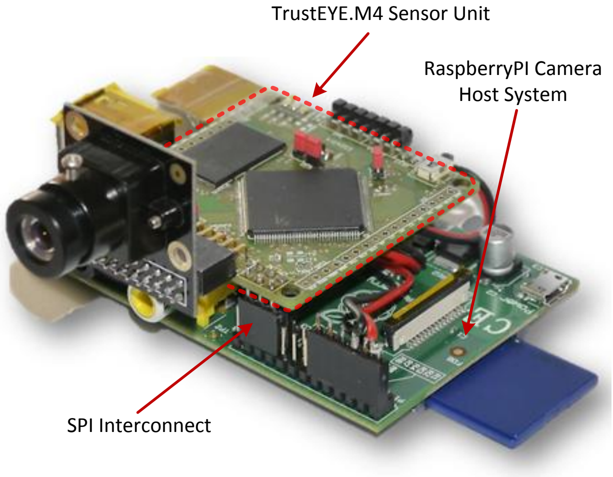

Figure 3: TrustEYE.M4 used as a secure sensing unit on top of a RaspberryPI Linux system serving as camera host system.

	ext. SRAM	int. SRAM
Mean Shift	89 ms	62 ms
Roberts Cross	11 ms	n/a

Table 1: Execution times on TrustEYE.M4.

via SPI to a RaspberryPI[1] single-board computer running Linux which serves as camera host system.

The TrustEYE.M4 CPU provides hardware accelerators for cryptographic algorithms including AES256, SHA1, SHA256 and HMAC. Furthermore, the SoC provides a true random number generator and a 96-bit unique ID. The chip's program Flash memory can be both permanently read- and write protected. The on-board ST33TPM12SPI TPM chip provides RSA key generation (2048 bits), RSA signature creation and encryption, secure monotonic counters, remote attestation capabilities and comes with an endorsement key certificate. The TPM is the basis for non-repudiation guarantees for captured images based on TPM-protected, non-migratable 2048 bit RSA keys.

3.1 Cartooning Privacy Filter

To demonstrate the proposed concept of a secure sensing unit we implement a streaming application that follows the structure outlined in Figure 2. The TrustEYE.M4 sensing unit is strictly separated from the RasperryPI camera host system where the Linux operating system, the streaming application and potential user applications are executed. Security flaws in, e.g., the network stack of the Linux operating system may result in a security breach of the RaspberryPI camera host system. The isolated sensing unit, which is the only entity that has access to the raw image data, is not affected. To demonstrate the capabilities of TrustEYE.M4, YUV422 images are read from the sensor which are then protected by the privacy filter based on a global cartoon-like effect. The resulting, pre-filtered image data is then forwarded to the camera host system for further processing.

The basic idea of cartooning is to generate ”cartoons” which allow to recognize behaviors but hinder the identification of persons in the scene. The two key techniques for cartooning are color segmentation and edge enhancements, i.e., smoothing areas with moderate color variations to single colored areas and to enhance important areas with emphasized edges. Mean shift filtering has been shown to deliver attractive results for color segmentation [12]. However, established mean shift filtering implementations as, e.g., *pyrMeanShiftFilter()* from OpenCV, are too complex to be ported to the resource-constraint TrustEYE.M4. Therefore, we implemented a customized filter inspired by mean shift which operates on YUV422 images delivered by the sensor and heavily relies on integral images. In our demonstration we achieve 11 fps for an image resolution of 320 × 240 pixels including transmission of uncompressed images to the RaspberryPI and subsequent streaming via Ethernet. Figure 4 shows an example of a cartooned image from TrustEYE.M4. Technical details about the cartooning filter are available in [33], and sample videos are available on the TrustEYE website [32]. Table 1 presents the runtimes for the customized mean shift function and the Roberts cross edge detection. With the iterative computation approach, the integral images fit into internal SRAM which notably speeds up data access. This results in a mean shift runtime of 62 ms per frame. Including edge-enhancement, a runtime of 73 ms per frame is achieved resulting in a theoretical frame rate of 13.7 fps. Due to SPI bus limitations and network overheads this is reduced in practice to the already mentioned 11 fps.

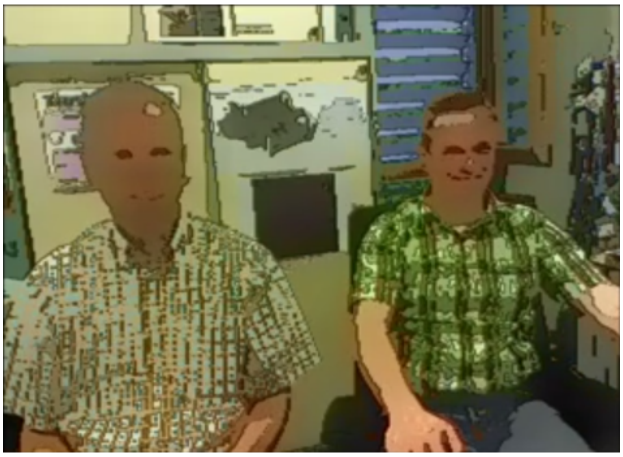

Figure 4: Cartooning image from the TrustEYE.M4 sensing unit.

4. CONCLUSION

In this paper we discussed the key security and privacy protection domains in smart camera networks and introduced our TrustEYE.M4 platform as an example for providing inherent security and privacy protection mechanisms as close as possible to the image sensor. We are confident that a holistic security and privacy protection approach is advantageous for a widespread deployment of smart cameras in smart environments.

5. ACKNOWLEDGMENTS

This work was performed as part of TrustEYE [32] and EPiCS and received funding from the European Commission (grant 257906), the European Regional Development Fund (ERDF) and the Carinthian Economic PromotionFund (KWF) under grant KWF-3520/23312/35521.

6. REFERENCES

[1] H. Aghajan and A. Cavallaro, editors. *Multi-Camera Networks: Principles and Applications.* Elsevier, 2009.

[2] P. K. Atrey, W.-Q. Yan, and M. S. Kankanhalli. A Scalable Signature Scheme for Video Authentication. *Multimedia Tools and Applications*, 34(1):107–135, 2006.

[3] A. Boukerche and D. Turgut. Secure Time Synchronization Protocols for Wireless Sensor Networks. *IEEE Wireless Communications*, 14(5):64–69, 2007.

[4] J. Brassil. Using Mobile Communications to Assert Privacy from Video Surveillance. In *Proceedings of the Parallel and Distributed Processing Symposium*, page 8, 2005.

[5] C. Castelluccia, A. C.-F. Chan, E. Mykletun, and G. Tsudik. Efficient and provably secure aggregation

[1] RaspberryPI Single Board Computer: http://www.raspberrypi.org (visited: 06/2014)

of encrypted data in wireless sensor networks. *ACM Transactions on Sensor Networks*, 5(3):1–36, May 2009.

[6] A. Cavallaro. Adding Privacy Constraints to Video-Based Applications. In *Proceedings of the European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology*, page 8, 2004.

[7] A. Cavallaro. Privacy in Video Surveillance. *IEEE Signal Processing Magazine*, 24(2):168–169, 2007.

[8] H. Chan, A. Perrig, and D. Song. Secure Hierarchical In-Network Aggregation in Sensor Networks. In *Proceedings of the International Conference on Computer and Communications Security*, pages 1–10, 2006.

[9] A. Chattopadhyay and T. E. Boult. PrivacyCam: A Privacy Preserving Camera Using uClinux on the Blackfin DSP. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[10] S.-C. S. Cheung, J. K. Paruchuri, and T. P. Nguyen. Managing Privacy Data in Pervasive Camera Networks. In *Proceedings of the International Conference on Image Processing*, pages 1676–1679, 2008.

[11] T. Dierks and C. Allen. RFC 2246: The TLS Protocol. `http://www.ietf.org/rfc/rfc2246.txt`, 1999.

[12] A. Erdélyi, T. Barát, P. Valet, T. Winkler, and B. Rinner. Adaptive Cartooning for Privacy Protection in Camera Networks. In *Proceedings of the International Conference on Advanced Video and Signal Based Surveillance*, page 6, 2014.

[13] Y.-C. Hu, A. Perrig, and D. B. Johnson. Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks. *Wireless Networks*, 11(1-2):21–38, Jan. 2005.

[14] C. Karlof and D. Wagner. Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures. *Ad Hoc Networks*, 1(2-3):293–315, 2003.

[15] M. Manzo, T. Roosta, and S. Sastry. Time Synchronization Attacks in Sensor Networks. In *Proceedings of the Workshop on Security of Ad Hoc and Sensor Networks*, page 10, 2005.

[16] I. Martínez-Ponte, X. Desurmont, J. Meessen, and J.-F. Delaigle. Robust Human Face Hiding Ensuring Privacy. In *Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services*, page 4, 2005.

[17] P. Meerwald and A. Uhl. Watermarking of Raw Digital Images in Camera Firmware. *IPSJ Transactions on Computer Vision and Applications*, 2:16–24, 2009.

[18] S. P. Mohanty. A Secure Digital Camera Architecture for Integrated Real-Time Digital Rights Management. *Journal of Systems Architecture*, 55(10-12):468–480, Oct. 2009.

[19] S. Ozdemir and Y. Xiao. Secure data aggregation in wireless sensor networks: A comprehensive overview. *Computer Networks*, 53(12):2022–2037, Aug. 2009.

[20] A. Perrig, R. Canetti, J. D. Tygar, and D. Song. The TESLA Broadcast Authentication Protocol The TESLA Broadcast Authentication Protocol. *RSA Cryptobytes*, (5), 2002.

[21] F. Z. Qureshi. Object-Video Streams for Preserving Privacy in Video Surveillance. In *Proceedings of the International Conference on Advanced Video and Signal-Based Surveillance*, pages 442–447, 2009.

[22] S. M. M. Rahman, M. A. Hossain, H. Mouftah, A. E. Saddik, and E. Okamoto. A Real-Time Privacy-Sensitive Data Hiding Approach based on Chaos Cryptography. In *Proceedings of the International Conference on Multimedia and Expo*, pages 72–77, 2010.

[23] M. Reisslein, B. Rinner, and A. Roy-Chowdhury. Smart Camera Networks. *Computer*, 47(5):26–28, May 2014.

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

[25] M. Saini, P. K. Atrey, S. Mehrotra, and M. S. Kankanhalli. Adaptive Transformation for Robust Privacy Protection in Video Surveillance. *Advances in Multimedia*, page 14, 2012.

[26] K. Sanzgiri, D. Laflamme, B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer. Authenticated Routing for Ad Hoc Networks. *IEEE Journal on Selected Areas in Communications*, 23(3):598–610, 2005.

[27] M. Schaffer and P. Schartner. Video Surveillance: A Distributed Approach to protect Privacy. In *Proceedings of the International Conference on Communications and Multimedia Security*, pages 140–149, 2007.

[28] D. N. Serpanos and A. Papalambrou. Security and Privacy in Distributed Smart Cameras. *Proceedings of the IEEE*, 96(10):1678–1687, Oct. 2008.

[29] D. Wagner. Resilient Aggregation in Sensor Networks. In *Proceedings of the Workshop on Security of Ad Hoc and Sensor Networks*, page 10, 2004.

[30] D. Westhoff, J. Girao, and M. Acharya. 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, 2006.

[31] J. Wickramasuriya, M. Datt, S. Mehrotra, and N. Venkatasubramanian. Privacy Protecting Data Collection in Media Spaces. In *Proceedings of the International Conference on Multimedia*, pages 48–55, 2004.

[32] T. Winkler. TrustEYE Project Website. `http://trusteye.aau.at`, 2012. last visited: June 2014.

[33] T. Winkler, A. Erdélyi, and B. Rinner. TrustEYE.M4: Protecting the Sensor – not the Camera. In *Proceedings of the International Conference on Advanced Video and Signal Based Surveillance*, page 6, 2014.

[34] T. Winkler and B. Rinner. Securing Embedded Smart Cameras with Trusted Computing. *EURASIP Journal on Wireless Communications and Networking*, 2011:20, 2011.

[35] T. Winkler and B. Rinner. Security and Privacy Protection in Visual Sensor Networks: A Survey. *ACM Computing Surveys*, 47(1):42, 2014. (in print).