

Digital Rights Management for Video Sensor Network

Taojun Wu, Liang Dai, Yuan Xue, Yi Cui
Department of Electrical Engineering and Computer Science
Vanderbilt University
Nashville, TN, 37235
{taojun.wu,liang.dai,yuan.xue,yi.cui}@vanderbilt.edu

Abstract

Video sensor network is evolving from an isolated system to an integral component of the global information infrastructure. In this paper, we argue that when video sensor network becomes a public information source on the Internet, DRM (Digital Rights Management) must be enforced, due to the sensitivity and the privacy natures of sensor content. Moreover, existing DRM solutions do not suffice because the explicit one-to-one mapping between content producer and consumer does not apply in the sensor network domain. We propose a DRM-enabled content service architecture for video sensor network. Within this architecture, we propose a binary-tree-based hierarchical key generation scheme for data encryption, and adopt a label-guided watermarking strategy to address the unique challenges of video sensor content. We present the evaluation results of our solution based on a preliminary video sensor testbed system.

1 Introduction

Sensor networks have dramatically changed the way people interact with physical world. They are deployed in a physical field collaborating to perform tasks from collecting information such as temperatures and real-time video images to locating the positions of tracking objects. In video sensor networks [4, 13, 11], each sensor is equipped with a camera which can provide important visual information. This useful technology has broad application in the fields of environmental monitoring, health care, military operations, and security surveillance, etc.

While most of the research efforts have been focused within the video sensor network itself, such as image and video compression [20, 14], video quality provisioning [10], bandwidth management [23] and energy efficiency [21, 5, 22], very few have addressed the fact that the data sink marks not the end, but the beginning of the journey of the video sensor content. Sensor network is now

at the crucial stage of evolving from an isolated system to an integral component of the global information infrastructure. The content collected by sensor systems not only holds practical values to individuals running them, but also can potentially benefit many other users. For example, a video sensor system monitoring the garage of a shopping mall is setup for security purposes. However, the archived video footage can become valuable materials for studies on customer shopping behaviors. When sensor network becomes a public information source on the Internet, many urgent technical issues arise, mainly due to the sensitivity and the privacy natures associated with the sensor content. In this paper, we argue for the crucial necessity of enforcing DRM (Digital Rights Management) in content servicing for the sensor network. Here, DRM refers to a collection of technologies used to handle the description, valuation, trading and monitoring of the rights held over any digital entity. There are many advantages to provide DRM to sensor contents. We list the most two salient ones as below.

- *Protecting Privacy*: In principal, DRM can effectively isolate its protected data from unauthorized access. For example, in patient monitoring, the access to any footage including the patient's appearance should be absolutely limited to his care-takers and family members.
- *Promoting Economical Incentives*: Many types of sensor content hold significant commercial values, such as highway traffic monitoring. DRM provides a set of solutions for the trading, accounting, and transaction processing of digital contents as commodities. Furthermore, digital forensic mechanisms are also available to identify and attest the abuse of access right and illegal distribution of the sensor content.

DRM has been proven technically sound in protecting digital work copyrights in movie and music industries[19]. Mature DRM systems have also been developed[12][3]. However, many intrinsic difficulties arise when deploying existing DRM solutions into the sensor network domain.

The challenge comes from the distinguishing data characteristics of the traditional digital content and sensor content. In typical DRM applications, an explicit one-to-one mapping exists between the producer and the consumer of the digital content, such as movies and music titles. Essentially a binary file, each piece of content is encrypted by a unique secret key prepared by its producer, i.e., the owner and distributor. End users, as the content consumer, must purchase a license that contains this key, in order to enjoy the content. Furthermore, the user’s access to a piece of content is all-or-nothing, e.g., an interested user must gain access to the movie by its entirety, not any of its subset.

Such a one-to-one mapping vanishes in the domain of sensor network. First, the sensor content is the spatial and temporal composition of data inputs from all sensors in the network. With respect to the information provided, the data streams produced by different sensors are often co-dependent. From the viewpoint of end users, what a meaningful piece of content (e.g., temperature in the playground) embodies is clearly detached from how it is produced (which sensors collectively created this result). Second, a user’s view towards the sensor content is often partial and customized due to factors like user interest and privacy protection. For example, in home monitoring for patient care, a video sensor network collects footage of patients within a geographical region. The care-taker of a patient may choose to view his/her activity during a certain period of time, but is clearly forbidden to view the footage of other patients within the same network.

In light of these challenges, any DRM solution for the sensor network must have extreme built-in flexibility during the collection, preparation, and access of sensor content. One might try to leverage existing DRM solution by dividing the data sampled by each sensor into numerous atomic units and treating each of them as an individual object like a movie file. For example, in video sensor network, if we set macro-block as the atomic unit, then any customized content requested by the user can be simply viewed as a collection of such units. Seemingly sound, this approach requires tremendous key space to keep up with the number of atomic units continuously produced by the sensor network. It also poses unacceptable key management overhead to the end user, whose requested content can easily result into thousands of units, hence the same number of keys. Therefore, we must propose a novel DRM solution, which effectively balances the trade-off among flexibility for content management, management overhead for content servicing, and usability for end users.

We propose a DRM-enabled content service architecture for video sensor network. Within this architecture, we propose a binary-tree-based hierarchical key generation scheme for data encryption, and adopt a label-guided watermarking strategy to address the unique challenges of sensor

content.

The rest of this paper is organized as follows. Sec. 2 overviews the video content service architecture and illustrates its key entities. Sec. 3 presents the digital right management framework and its security components for video sensor networks. Sec. 4 presents our evaluation results obtained from our preliminary testbed system. We finally conclude at Sec. 6.

2 Sensor Content Service Architecture

In this section, we present the architecture of video sensor content service. Fig. 1 outlines the flow of sensor content from its creation, collection, to distribution. In this picture, we identify three entities, *content provider*, *service provider* and *end user*.

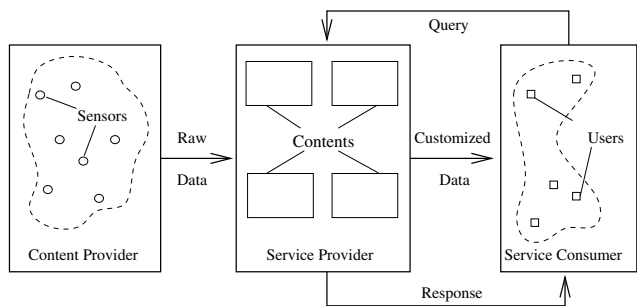


Figure 1. Sensor Content Service Architecture

- *Content Provider (CP)* deploys video sensor networks and collects raw data from all these sensors.
- *End User (EU)* (service consumer) is requests a subset of sensor content according to individual interest.
- *Service Provider (SP)* acts as interface between *CP* and *EU*. *SP* processes the heterogeneous sets of raw data from different *CP*s, decomposes and transforms them into sensor content with unified format. The sensor content is customized according to users’ requests. When *EU*’s request arrives, *SP* parses it and responds back with customized content subset.

Defining an interface to map between the raw sensor data and the content customizable by the end user is the main purpose to implement this service architecture. The main challenge, as we have articulated in the introduction, comes from the reality that the explicit one-to-one mapping between the content producer and consumer in traditional DRM applications does not apply for the domain of video

sensor network. In what follows, we illustrate the functioning of the above entities in the context of video sensor network.

We start with the raw image data/video data collected by the video sensors. Let S_i ($i = 1, 2, \dots, N$) be one of the N video sensors in the network. The data content is provided by S_i as a content stream $ConStream_i$ across a three dimensional domain (one dimension in temporal domain and two dimensions in spatial domain). In the temporal domain, $ConStream_i$ consists of a series of content items ($ConItem_i(t), t \in [t_i^1, t_i^2]$). In the spatial domain, each $ConItem_i(t)$ (or video frame) is decomposed into small content units ($ConUnit_i^j(t)$), which are the smallest content units to respond to user-specific queries. A $ConUnit$ is part of a video frame and is the smallest element for encryption at the content server of CP . A choice of encryption at this granularity serves two purposes: to save only useful information and to support customization of EUs .

The relationship of these three units is illustrated in Fig. 2. Mathematically, we have $ConUnit_i^j(t) \in ConItem_i(t)$, $ConItem_i(t) \in ConStream_i$ and $\bigcup_{i \in [1, N]} ConStream_i$ constitutes the whole set of content provided by CP .

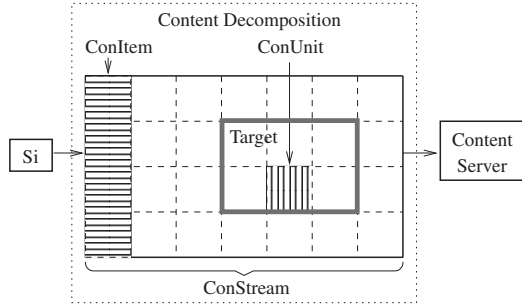


Figure 2. Sensor Content Decomposition

From a user’s perspective, the unit of interested content is defined as a *target*. In video sensor network, a target can be home, highway, garage, etc. When an EU requests the desired target from sensor content service, he/she will submit a profile of the target which may include its identity, position, size and time interval of the request. Based on the profile, the SP maps the target into a set of content units, which collectively embody this target. The set of content units are then delivered to the EU .

For example, a target’s profile can be denoted as $\{ID = \text{“BankFrontDoor”}, POS = (5, 7), SIZE = (9, 15), TIME = [100s, 120s]\}$. This maps to a set of images $Img_i(t)$ with region size 9 by 15 (in pixels) at position (5,7) during time t , $100s \leq t \leq 120s$.

3 Digital Right Management for Video Sensor Networks

Based on the established video sensor content service architecture, we present our DRM-safe framework and its security components. The most salient feature of DRM is the remote control over the digital rights of content. The contents collected by CP are delivered to SP under protection of existing link-level encryption protocols in Sensor Network domain, such as TinySec [9]. Such light-weighted encryption is used at the video sensors due to their limited resources. At the SP where the sensor content is decomposed, a more robust encryption will be applied to withhold longer term attacks on the stored sensor content at SP . The end users will then need to acquire the secret key from the SP to decrypt the content. This solution imposes no security requirement over the content distribution channel. It also avoids the overhead of repetitive on-demand encryption over large volume of multimedia data. Further digital watermarks are embedded into the video content to discourage the end users to further disseminate the content.

In this paper, we present two security mechanisms employed at SP in our DRM framework which include (1) data encryption, which protects the confidentiality of the digital content, and (2) sensor content and end user association via digital watermarking, which provides forensic means to prevent leaking of the digital content, via embedding user-specific information into the data to be distributed.

While existing DRM solutions have well addressed these challenges, they cannot be applied to the domain of video sensor networks directly. When accessing the content from a video sensor network, the user requests can be highly heterogeneous across spatial and temporal domains, at varying granularity. In our content service architecture, some requests may only involve a handful of content units, while others may cover thousands of them. Obviously, it is not realistic to find a one-size-fit-all solution by looking for the right size of the content unit. Similarly, this challenge also applies to the practice of digital watermarking on the requested sensor content.

To address the above challenges, we present a *tree based hierarchical key management and encryption scheme* for data encryption, and adopt a *label-guided watermarking strategy* for digital watermarking.

3.1 Hierarchical Key Management and Data Encryption

Key Generation: The design goal of key management at the SP is to support a scalable data encryption solution that is adaptive to highly heterogeneous sensor content requests. The basic idea is to generate a hierarchical key structure corresponding to the content item structure. The keys at

the lower level of the hierarchy could be generated from the keys at the higher level. For each content unit (*ConUnit* – the smallest unit that respond to users’ requests), the keys at the lowest level (leaf keys) are used for encryption. When a content stream (*ConStream*) that consists of multiple *ConUnits* is requested, instead of providing all the leaf keys that encrypt these *ConUnits* to the end user, only the keys which constitute the minimum cover of these keys in the tree hierarchy are provided.

Specifically, the *CP* and *SP* reach an agreement about a common provider’s master key $MasterKey_P$ first. This $MasterKey_P$ is the highest-level key in the hierarchy and will be used throughout their sensor network content provision contract. For every sensor S_i , *SP* generates a sensor master key $MasterKey_i$ using hash function with S_i ’s profile and provider’s master key as input. All $MasterKey_i$ s are updated on a regular basis.

$$MasterKey_i = HASH(SensorProfile_i || MasterKey_P) \quad (1)$$

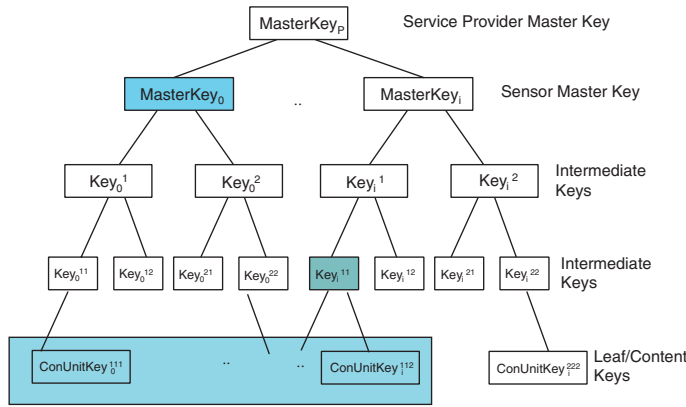


Figure 3. Hierarchical Key Generation

For each sensor S_i , its $MasterKey_i$ is used to generate content unit keys ($ConUnitKey_i^j$) through a tree-based key hierarchy as shown in Fig. 3. At each level k , we consider intermediate key with root node $Key_i^0 = MasterKey_i$; and leaf nodes as content unit keys $ConUnitKey_i^j$.

$$Key_i^{(k-1)l} = HASH(l || Key_i^{k-1}) \quad (2)$$

Here, l represents the “tree node position” which could be time range, or the region position of the image. At time t , the content unit ($ConUnit_i^l(t)$) will be encrypted with the content unit key ($ConUnitKey_i^j$) corresponding to the leaf node. The License Server in *SP* keeps a log of all master, intermediate and content keys ever generated and used within the DRM system.

Encryption: Note that when the sensor content moves from *CP* to *SP*, they are encrypted by link-level proto-

cols like TinySec. At the *SP*, sensor content is decomposed into content units and further encrypted by its corresponding content keys with conventional symmetric ciphers. Upon end users’ request, content units of targets resulting from decomposition of the *EU* request will be delivered encrypted.

The same *ConUnits* of $ConStream_i$ from one sensor S_i will be encrypted with one single key for a certain period, as described in key generation step. Hence, potential adversary possessing access privilege to one target content cannot decipher other unlawfully acquired target contents, even if they are all from the same sensor source. The periodically updating characteristic of encryption key of the same target will ensure that a user is granted only limited time length access rights to his desired target. To be able to access content over a larger time domain, the *EU* will need to request appropriate access right through multiple licenses.

Access Control: Digital right management enables the sensor data content to be delivered to end users via diverse, non-secure communication channels. To actually playback the received encrypted contents, the *EU* will need to authenticate himself and request appropriate access right for the sensor contents. To do so, the *EU* will request a license from the *SP*. In his request, the *EU* will indicate for what content and time interval he is requesting a license. Also included will be his personal information (e.g. unique ID or credit card number, if payment is necessary).

After user authentication, *SP* will verify his access right to the requested sensor content units and generate a license. The license will include all the keys (intermediate and content keys as will be detailed soon) required to decrypt the content. In our tree-based key hierarchy, this corresponds to a minimum cover of all the leaf keys that correspond to the content units requested by the user. As illustrated in Fig. 3, if the *EU* will request all content units as circled, then instead of returning all keys associated with each unit, the *SP* only needs to return the keys of the shadowed nodes, which is $MasterKey_0$ and Key_i^{11} . From the key generation procedure, it is obvious that the user is able to derive all content keys required to decrypt the content he requested. To protect the confidentiality and integrity of the license, it will be signed with *SP*’s private key by License Server and encrypted via *EU*’s public key.

Our hierarchical key management solution scales well to user requests for large volume contents. It clearly reduces the overhead and complexity involved in communicating the keys to the end users. At the same time it is also flexible enough to meet the diverse user requests for sensor content with different sizes (number of content units). The key hierarchy doubles the key space that the *SP* needs to manage, in comparison with a flat key management solution where content unit keys are organized in a flat way. Yet as *SP* usually resides on powerful servers, such increase

would not significantly affect the performance of our DRM framework for video sensor content service.

3.2 Legal EU-Sensor Content Association

Watermarking [16, 6] is the process of embedding data into a multimedia content such as image, audio and video. The embedded information, called watermark can be extracted later on for security reasons. In our DRM framework, digital watermarking of the generated sensor content at *CP* and *SP* is used to (1) protect the rightful ownership of *CP* and *SP*; (2) discourage *EUs* from abusing their digital rights and enables *CP* and *SP* to trace illegal sensor content distributions and identify leakers.

In particular, *SP* will prepare a composed sensor content consisting of individual sensor content units, desired and requested by the *EU*. First, the *CP* and *SP* will generate DRM-safe sensor content, where each sensor content unit will carry the hierarchical watermark consisting of *CP* and *SP* rightful ownership information. However, this composed watermark will not be sufficient to create a sensor content stream that would be unambiguous for different *EUs*. To address this issue, we present a third watermarking process laid over sensor content – *label-guided watermarking scheme*, which is able to provide efficient yet powerful digital watermarking for unambiguous and legal sensor content association with a *EU/customer*.

The label-guided watermarking scheme has the following steps: First, the *SP* chooses two watermarks W_0 and W_1 . And let $S^j = (S_1^j, \dots, S_n^j)$ be the composed sensor content stream prepared for an *EU* j . As the next step, the *SP* copies the stream S^j to create $S^{j'}$ stream and watermarks the sensor content stream S^j with watermark W_0 and stream $S^{j'}$ with watermark W_1 . Then the *SP* will generate a unique *EU/customer* label in form of a *binary identification key* b (e.g., 011010100). Such a label is generated by a hash function based on the information of the user's request (e.g., user's ID, query content, etc.)

$$b = \text{HASH}(\text{UserID} \parallel \text{UserQuery} \parallel \text{TimeStamp}) \quad (3)$$

This generated label string will be used to determine which watermark each outgoing content unit should have. Finally, a watermarked composed sensor content stream S_{final}^j which is unique to the *EU* is generated as follows.

- If the identification key binary digit will be 0, then a sensor content unit from the stream S^j watermarked with watermark W_0 will be selected;
- If the identification key binary digit will be 1, content unit from $S^{j'}$ with watermark W_1 will be selected.

It means that the generated label string will be able to determine which watermark each outgoing content unit should have¹. That is, depending on if b_k is 0 or 1, the k th content unit has watermark W_0 or W_1 . In this way, the combination of two different watermarks in content units reveals the source (one content service at a particular time to a particular user) of leaked digital contents. The process of label guided content service is shown in Fig. 4. If the number of content units in a single request exceeds $|b|$, the label will be scanned from left to right repeatedly until all content units are dispatched.

For example, let us assume

$$S = (\text{Content}_{S_1}^{W_0}, \text{Content}_{S_2}^{W_0}, \text{Content}_{S_3}^{W_0}, \text{Content}_{S_4}^{W_0}), \quad (4)$$

$$S_{copy} = (\text{Content}_{S_1}^{W_1}, \text{Content}_{S_2}^{W_1}, \text{Content}_{S_3}^{W_1}, \text{Content}_{S_4}^{W_1}) \quad (5)$$

and the content units are coarsely partitioned at the level of different sensor information (note that the granularity in reality will be much finer, going into small ConUnit_i^j units). If the *EU* identification key would be 0110, then the resulting sensor content stream for the *EU* would be

$$S_{final} = (\text{Content}_{S_1}^{W_0}, \text{Content}_{S_2}^{W_1}, \text{Content}_{S_3}^{W_1}, \text{Content}_{S_4}^{W_0}). \quad (6)$$

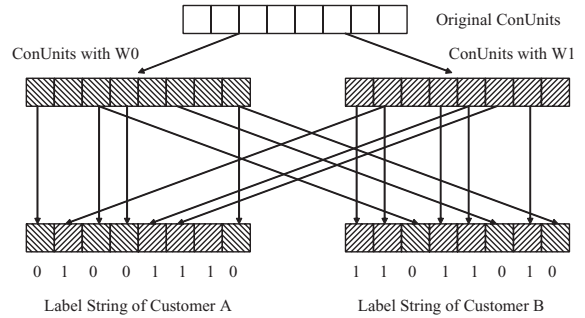


Figure 4. Label-Guided Content Servicing

Utilizing the label-based watermarking we are able to study the attacks, collusion possibilities among *EUs*, and leaking actions. Due to the fact that a single content unit, or a small collection of contents will be usually meaningless, and that locating such small-scale actions will be extremely difficult, we assume a content breach action to be one that leaks out at least $|b|$ content units. Suppose a subset (denoted as I_B) of m content units are breached, and a series of bits (0, 1) denote whether each content unit is watermarked with W_0 or W_1 . We can arrange these bits into a unique

¹Assume that all $\text{ConUnit}_i^j(t)$ s are totally ordered.

Breach String $BStr$ of binary bits b_1, b_2, \dots, b_m following their corresponding $ConUnits$ ' order. By inspecting $BStr$, we can identify a repeating substring $subBStr$ of length $|b|$. This $subBStr$ will uniquely identify the leaker.

3.3 Overall DRM Framework

Integrating the above pieces together, the overall DRM framework is shown in Fig. 5. The four important components of DRM that will be implemented at SP are: Content Server, Query Server, Policy Server and License Server. Query Server will parse EU 's queries. Policy Server will be responsible for sensor data content access control. License Server will keep track of all past and present encryption keys. It will also deal with license requests from EU and record granted access rights of individual users. Content Server will store watermarked and encrypted content units and dispatches requested contents to EU in designated way. The Encryption/SP Watermarking component will provide security functionalities like encryption, message authentication, digital signature, license generation, label-based watermarking. On EU side, the DRM Manager will take charge of requesting and verifying licenses and enforcing digital right check. The Decryption component will decrypt contents for the Content Player.

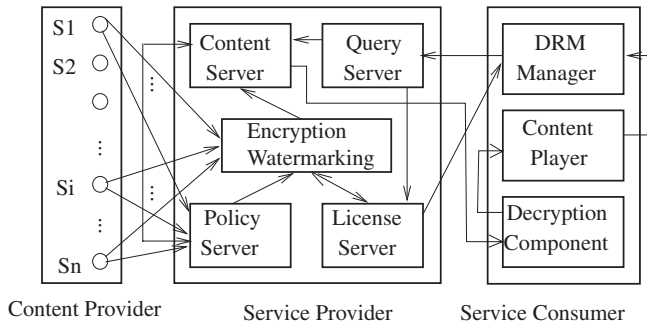


Figure 5. Digital Right Management in Wireless Sensor Network

Our framework is viable under several attack scenarios. For typical eavesdropping attacks, even if an attacker can intercept the video contents sent to an $EU A$, he can not possibly acquire all necessary keys to decrypt them. Furthermore, illegal distribution is countered as well. Any valid $EU A$ is identified by a unique $UserID_A$, which relates to his *binary identification key*. In case of content breach by B , the breach string will identify him out.

4 Experimental Results

4.1 Testbed Setup

We build our testbed system on Linux system (Dell Precision 670, dual-core, 2GB RAM). We prepare a set of webcam images (average size 225KB) to simulate the video sensor content. These images have a unified resolution of 320×240 . Each image is split into several $ImgRegs$, each representing a target. Every $ImgReg$ is duplicated to two copies, watermarked with different keys. A unique encryption key is generated separately for each $ImgReg$. In the preliminary experiments reported in this paper, we evaluate the average performance of watermarking, encryption, and decryption on each $ImgReg$.

Our watermarking experiment uses the watermarking scheme provided by the Digital Invisible Ink Toolkit (DIIT)[1]. The encryption experiment is built upon the Java security library. Furthermore, we rely on the Message Digest feature to produce unique encryption key for each target. The encryption algorithms we choose are DES and RC4.

Fig. 6 illustrates the flow of our experiments. Fig. 6 (a) shows a webcam image of the engineering campus of Vanderbilt University. The original image is then split into four $ImgRegs$, with each watermarked individually. Fig. 6 (b) shows the watermark results of the bottom two $ImgRegs$. The same $ImgRegs$ after encryption are shown in Fig. 6 (c). Finally, in Fig. 6 (d), we show client-side result of a user interested to monitor solely the Small Molecule NMR Facility Core, which is the round building shown in the lower left $ImgReg$ of the original image. While this $ImgReg$ is decrypted by acquiring the corresponding decryption key, the remaining $ImgReg$ remains encrypted to the viewer.

4.2 Evaluation Results

4.2.1 Watermark Size Effect

Fig. 7 demonstrates the time overhead of watermarking on the same image using watermarks of different sizes. Also using bar graph in the same figure, we show the ratio of watermark size to $MaxBytes$, the maximum bytes that can be hidden in an image. According to DIIT, it can be calculated as $MaxBytes = (ImageHeight \times ImageWidth \times Color\ Numbers \times Number\ of\ Bits\ to\ Hide)/8$. As shown in the picture, the time overhead does not grow linearly as message size does. This is because as the ratio becomes larger, it takes the watermarking program longer time to find the free space and hide the information.

Fig. 8 compares results of watermarking two images of different sizes using the same set of watermarks. The smaller image is 6.9KB, and the larger one is 20.8KB. The

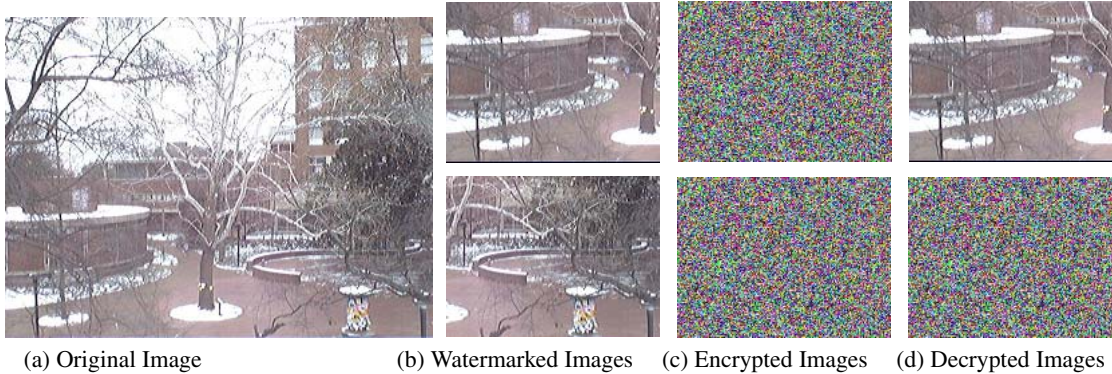


Figure 6. Comparison of original, watermarked, encrypted and decrypted images

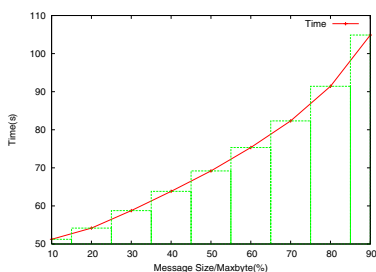


Figure 7. Comparison of time cost with different message sizes.

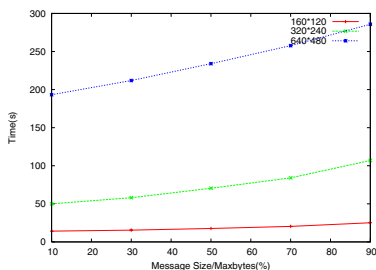


Figure 8. Comparison of time cost of different images with varying message sizes.

sizes of message range from 1KB to 24KB. In the picture, the two curves are almost overlapped. Although the large image does take longer time than the smaller one, the difference is trivial. Combined with the results in Fig. 7, the size of watermarks has greater impact than the size of the image.

4.2.2 Key Generation Performance

We test the time to create a finite number of keys. The result shows that it takes 150ms to create 100 keys but takes

only about 450ms to create 10,000 keys. Since we just run the key creation function repeatedly, and did not save created key into the storage system, the result may be over-optimistic. But even for 150ms/100 keys, it can still support a system with large key number requirement.

5 Related Work

Our work relies on extensive prior work in Digital Rights Management (DRM) in distributed multimedia systems and research results in several areas including watermarking encryption algorithms and security protocols.

Major players in Internet-based multimedia have adopted DRM into their mainstream products. Windows Media DRM [3] is a flexible platform that makes it possible to protect and securely deliver content by subscription or individual request. Developed by RealNetworks, Helix DRM [2] is a comprehensive and flexible platform for the secure media content delivery of standards-based as well as leading Internet formats, including RealAudio, RealVideo, MP3, MPEG-4, AAC, H.263, etc. Both solutions provide secure media packaging, license generation and content delivery to a trusted media player on a computer, portable device, or network device. DRM has also been applied in preserving the privacy of user context information in ubiquitous computing environment [8]. However, none of the existing DRM solution could be applied to protect the video sensor content due to the challenges we have presented. In [18, 15], two hierarchical access control and key management frameworks are presented. Our work is different from [18, 15] in that we consider the unique temporal and spatial diversity characteristics of video sensor contents.

Security for wireless sensor networks has been extensively studied in the existing literature, which includes link layer security [9], broadcast authentication [17], key management [7]. Although both concern the security issues involved with the emergence of sensor networks, the existing research has focused on protecting the information within

sensor networks, our work mainly focuses on the preserving the privacy and economical value of the sensor information when it is delivered from sensor networks to the Internet.

6 Conclusion

Digital right management is a critical component to enable the vision of sensor-centric global information infrastructure. This paper presents the architecture and the enabling security mechanisms of digital right management for video sensor networks. Novel key management scheme is presented to address the unique challenge of video sensor data content distribution. Initial testbed results show that our proposed solution is sound and efficient. We will expand our experiment from single images to continuous streams as future work.

7 Acknowledgement

We thank Professor Klara Nahrstedt from the University of Illinois for insightful discussion on this paper. This work was supported in part by TRUST (The Team for Research in Ubiquitous Secure Technology), which receives support from the National Science Foundation (NSF award number CCF-0424422).

References

- [1] Digital invisible ink toolkit (diit). <http://diit.sourceforge.net>.
- [2] Helix drm. <http://www.realnetworks.com/products/drm/index.html>.
- [3] Windows media drm. <http://www.microsoft.com/windows/windowsmedia/drm/default.aspx>.
- [4] W. chi Feng, B. Code, E. Kaiser, W. chang Feng, and M. L. Baillif. Panoptes: Scalable low-power video sensor networking technologies. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 1(2):151–167, 2005.
- [5] C. Chiasserini and E. Magli. Energy consumption and image quality in wireless video-surveillance networks. In *Proc. of 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 2357–2361, September 2002.
- [6] I. J. Cox and M. Miller. A Review of Watermarking and the Importance of Perceptual Modeling. In *Proceedings of the IS&T/SPIE Conference on Human Vision & Electronic Imaging II*, volume 3016, pages 92–99, San Jose, CA, February 1997.
- [7] L. Eschenauer and V. D. Gligor. A key management scheme for distributed sensor networks. In *Proc. ACM CCS*, Washington, DC, 2002.
- [8] M. Fahrmaier, W. Sitou, and B. Spanfelner. Security and privacy rights management for mobile and ubiquitous computing. In *IEEE UbiComp*, 2005.
- [9] C. Karlof, N. Sastry, and D. Wagner. Tinysec: A link layer security architecture for wireless sensor networks. In *Proc. ACM SenSys*, 2004.
- [10] P. Korshunov and W. T. Ooi. Critical video quality for distributed automated video surveillance. In *ACM MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, 2005.
- [11] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu. Senseeye: a multi-tier camera sensor network. In *ACM MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, 2005.
- [12] E. Lin, A. Eskicioglu, R. Lagendijk, and E. Delp. Advances in digital video content protection. *Proceedings of IEEE*, 93(1):171–183, 2005.
- [13] L. Jiao, Y. Wu, G. Wu, E. Y. Chang, and Y. Wang. The anatomy of a multi-camera security surveillance system. *ACM Multimedia System Journal*, pages 144–163, October 2004.
- [14] E. Magli, M. Mancin, and L. Merello. Low complexity video compression for wireless sensor networks. In *Proc. of 2003 International Conference on Multimedia and Expo*, pages 585–588, July 2003.
- [15] G. Miklau and D. Suci. Controlling access to published data using cryptography. In *IEEE VLDB*, 2003.
- [16] M.S. Swanson, M. Kobayashi, and A. Tewfik. Multimedia embedding and watermarking technologies. In *Proceedings of IEEE*, volume 86(6), pages 1064–1088, June 1998.
- [17] A. Perrig, R. Szewczyk, J. Tygar, V. Wen, and D. Culler. Spins: Security protocols for sensor networks. In *Wireless Networks Journal*, Springer, volume 8(5), pages 521–534, September 2002.
- [18] R. S. Sandhu. Cryptographic implementation of a tree hierarchy for access control. *Inf. Process. Lett.*, 27(2):95–98, 1988.
- [19] M. Stamp. Risks of digital rights management. *Commun. ACM*, 45(9), 2002.
- [20] R. Wagner, R. Nowak, and R. Baraniuk. Distributed image compression for sensor networks using correspondence analysis and super-resolution. In *Proc. of International Conference on Image Processing (ICIP)*, pages 597 – 600, September 2003.
- [21] Y. Xue, Y. Cui, and K. Nahrstedt. Maximizing lifetime for data aggregation in wireless sensor networks. *ACM/Kluwer Mobile Networks and Applications (MONET) Special Issue on Energy Constraints and Lifetime Performance in Wireless Sensor Networks*, 2004.
- [22] W. Yu, Z. Sahinoglu, and A. Vetro. Energy efficient jpeg 2000 image transmission over wireless sensor networks. In *Proc. of Global Telecommunications Conference (GLOBECOM)*, pages 2738–2743, December 2004.
- [23] Z. Yang and K. Nahrstedt. A bandwidth management framework for wireless camera array. In *Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, 2005.