

Protecting and Managing Privacy Information In Video Surveillance Systems

M. V. Venkatesh, S.-C. S. Cheung, J. K. Paruchuri, J. Zhao and T. Nguyen

Abstract Recent widespread deployment and increased sophistication of video surveillance systems have raised apprehension on their threat to individuals' right of privacy. Privacy protection technologies developed thus far have focused mainly on different visual obfuscation techniques but no comprehensive solution has yet been proposed. We describe a prototype system for privacy-protected video surveillance that advances the state-of-the-art in three different areas: First, after identifying the individuals whose privacy needs to be protected, a fast and effective video inpainting algorithm is applied to erase individuals' images as a means of privacy protection. Second, to authenticate this modification, a novel rate-distortion optimized data-hiding scheme is used to embed the extracted private information into the modified video. While keeping the modified video standard-compliant, our data hiding scheme allows the original data to be retrieved with proper authentication. Third, we view the original video as a private property of the individuals in it and develop a secure infrastructure similar to a Digital Right Management system that allows individuals to selectively grant access to their privacy information.

1 Introduction

Rapid technological advances have ushered dramatic improvements in techniques for collecting, storing and sharing personal information among government agencies and private sectors. Even though the advantages brought forth by these methods cannot be disputed, the general public are becoming increasingly wary about the erosion of their rights of privacy [15]. While new legislature and policy changes are needed to provide a collective protection of personal privacy, technologies are play-

Except for T. Nguyen, all the authors are with the Center for Visualization and Virtual Environments, University of Kentucky, Lexington, KY 40507. T. Nguyen is with the School of Electrical Engineering and Computer Science at Oregon State University, Corvallis, OR 97331. Contacting author: Sen-ching S. Cheung e-mail: cheung@engr.uky.edu

ing an equally pivotal role in safeguarding private information [13]. From encrypting online financial transactions to anonymizing email traffic [12], from automated negotiation of privacy preference [17] to privacy protection in data mining [27], a wide range of cryptographic techniques and security systems have been developed to protect sensitive personal information.

While these techniques work well for textual and categorical information, they cannot be directly used for privacy protection of imagery data. The most relevant example is video surveillance. Video surveillance systems are the most pervasive and commonly-used imagery systems in large cooperations today. Sensitive information including identities of individuals, activities, routes and association are routinely monitored by machines and human agents alike. While such information about distrusted visitors is important for security, misuse of private information about trusted employees can severely hamper their morale and may even lead to unnecessary litigation. As such, we need privacy protection schemes that can protect selected individuals without degrading the visual quality needed for security. Data encryption or scrambling schemes are not applicable as the protected video is no longer viewable. Simple image blurring, while appropriate to protect individuals' identities in television broadcast, modifies the surveillance videos in an irreversible fashion, making them unsuitable for use as evidence in the court of law.

Since video surveillance poses unique privacy challenges, it is important to first define the overall goals of privacy protection. We postulate here the five essential attributes of a privacy protection system for video surveillance. In a typical digital video surveillance system, the surveillance video is stored as individual segments of fixed duration, each with unique ID that signifies the time and the camera from which it is captured. We call an individual *an user* if the system has a way to uniquely identify this individual in a video segment, using a RFID tag for example, and there is a need to protect his/her visual privacy. The imagery about a user in a video segment is referred to as *private information*. A *protected video* segment means that all the privacy information has been removed. A *client* refers to a party who is interested in viewing the privacy information of a user. Given these definitions, a privacy protection system should satisfy these five goals:

Privacy Without the proper authorization, a protected video and the associated data should provide no information on whether a particular user is in the scene.

Usability A protected video should be free from visible artifacts introduced by video processing. This criterion enables the protected video for further legitimate computer vision tasks.

Security Raw data should only be present at the sensors and at the computing units that possess the appropriate permission.

Accessibility A user can provide or prohibit a client's access to his/her imageries in a protected video segment captured at a specific time by a specific camera.

Scalability The architecture should be scalable to many cameras and should contain no single point of failure.

In this chapter, we present an end-to-end design of a privacy protecting video surveillance system that possesses these five essential features. Our proposed de-

sign advances the state-of-the-art visual privacy enhancement technologies in the following aspects:

1. To provide complete privacy protection, we apply video inpainting algorithm to erase privacy information from video. This modification process not only offers effective privacy protection but also maintains the apparent nature of the video making it usable for further data processing.
2. To authenticate this video modification task, a novel rate-distortion optimized data-hiding scheme is used to embed the identified private information into the modified video. The data hiding process allows the embedded data to be retrieved with proper authentication. This retrieved information along with the inpainted video can be used to recover the original data.
3. To provide complete control of privacy information, we view the embedded information as private property of the users and develop a secure infrastructure similar to a Digital Right Management system that allows users to selectively grant access to their privacy information.

The rest of the chapter is organized as follows: in Section 2, we provide a comprehensive review on existing methods to locate visual privacy information, to obfuscate video and to manage privacy data. In Section 3, we describe the design of our proposed system and demonstrate its performance. Finally in Section 4 we identify the open problems in privacy protection for video surveillance and suggest potential approaches towards solving them.

2 Related Works

There are three major aspects to privacy protection in video surveillance systems. The first task is to identify the privacy information needed to be preserved. The next step is to determine a suitable video modification technique that can be used to protect privacy. Finally, a privacy data management needs to be devised to securely preserve and manage the privacy information. Here we provide an overview of existing methods to address these issues and discuss the motivation behind our approach.

2.1 Privacy Information Identification

The first step in the privacy protection system is to identify individuals whose privacy needed to be protected. While face recognition is obviously the least intrusive technique, its performance is highly questionable in typical surveillance environments with low-resolution cameras, non-cooperative subjects and uncontrolled illumination [18]. Specialized visual markers are sometimes used to enhanced recognition. In [37], Schiff et al. have these individuals wearing yellow hard hats for

identification. An Adaboost classifier is used to identify the specific color of a hard hat. The face associated with the hat is subsequently blocked for privacy protection. While the colored hats may minimize occlusion and provide a visual cue for tracking and recognition, its prominent presence may be singled out in certain environments. A much smaller colored tag worn on the chest was used in our earlier work [50]. To combat mutual and self occlusion, we develop multiple camera planning algorithms to optimally place cameras in arbitrary-shaped environments in order to triangulate the location of these tags.

Non-visual modality can also be used but they require additional hardware for detection. Megherbi et al. exploit a variety of features including color, position, and acoustic parameters in a probabilistic frame to track and identify individuals [28]. Kumar et al. present a low-cost surveillance system employing multimodality information, including video, infrared, and audio signals, for monitoring small areas and detecting alarming events [26]. Shakshuki, et al. has also incorporated Global Positioning System (GPS) to aid the tracking of objects [39]. The drawback of these systems is that audio information and GPS signals are not suitable for use in indoor facilities with complicated topology.

Indoor wireless identification technologies such as RFID systems offer better signal propagation characteristics when operating indoors. Nevertheless, the design of a real-time indoor wireless human tracking system remains a difficult task [42] – traditional high frequency wireless tracking technologies like ultra-high frequency (UHF) and ultra-wideband (UWB) systems do not work well at significant ranges in highly reflective environments. Conversely, more accurate short-range tracking technologies, like infrared (IR) or ultrasonics, require an uneconomically dense network of sensors for complete coverage. In our system, we have chosen to use a wireless tracking system based on a technology Near-Field Electromagnetic Ranging (NFER[®]). NFER exploits the properties of medium and low-frequency signals within about a half wavelength of a transmitter. Typical operating frequencies are within the AM broadcast band (530-1710 kHz). The low frequencies used by NFER are more penetrating and less prone to multi-path than microwave frequencies. These near-field relationships are more fully described in a patent [36] and elsewhere [35]. In our system, each user wears an active RFID tag that broadcasts a RF signal of unique frequency. After triangulating the correspondence between the RF signals received at three antennas, the 2-D spatial location of each active tag can then be continuously tracked in real-time. This location information, along with the visual information from the camera network is combined to identify those individuals whose privacy needs to be protected.

It should be pointed out that there are privacy protection schemes that do not require identification of privacy information. For example, the PrivacyCam surveillance system developed at IBM protects privacy by revealing only the relevant information such as object tracks or suspicious activities [38]. While this may be a sensible approach for some applications, such a system is limited by the types of events it can detect and may have problems balancing privacy protection with the particular needs of a security officer.

2.2 *Privacy Information Obfuscation*

Once privacy information in the video has been identified, we need to obfuscate them for privacy protection. There are a large variety of such video obfuscation techniques, ranging from the use of black boxes or large pixels (pixelation) in [2, 45, 37, 8] to complete object replacement or removal in [19, 30, 48, 44]. Black boxes or pixelation has been shown to be inadequate in fully protecting a person's identity [30]. Moreover these kinds of perturbations to multimedia signals destroy the nature of the signals, limiting their utility for most practical purposes. Object replacement techniques gear at replacing sensitive information such as human faces or bodies with generic faces [30] or stick figures [44] for privacy protection. Such techniques require precise position and pose tracking which are beyond the reach of current surveillance technologies. Cryptographical techniques such as secure multi-party computation have also been proposed to protect privacy of multimedia data [1, 21]. Sensitive information is encrypted or transformed in a different domain such that the data is no longer recognizable but certain image processing operations can still be performed. While these techniques provide strong security guarantee, they are computationally intensive and at the current stage, they support only a limited set of image processing operations.

We believe that complete object removal proposed in [19, 9] provides a more reasonable and efficient solution for full privacy protection while preserving a natural-looking video amenable to further vision processing. This is especially true for surveillance video of transient traffic at hallways or entrances where people have limited interaction with the environment. The main challenge with this approach lies in recreating occluded objects and motion after the removal of private information. We can accomplish this task through video inpainting which is an image processing technique used to fill in missing regions in a seamless manner. Here we briefly review existing video inpainting and outline our contributions in this area.

Early work in video inpainting focus primarily on repairing small regions caused by error in transmission or damaged medium and are not suitable to complete large holes due to the removal of visual objects [4, 3]. In [46], the authors introduce the Space-Time video completion scheme which attempts to fill the hole by sampling spatio-temporal patches from the existing video. The exhaustive search strategy used to find the appropriate patches makes it very computationally intensive. Patwardhan et al. extend the idea of prioritizing structures in image inpainting in [11] to video [32]. Inpainting techniques that make use of the motion information along with texture synthesis and color re-sampling have been proposed in [49, 40]. These schemes rely on local motion estimates which are sensitive to noise and have difficulty in replicating large motion. Other object-based video inpainting such as [23] and [24] rely on user-assisted or computationally intensive object segmentation procedures which are difficult to deploy in existing surveillance camera networks.

Our approach advocates the use of semantic objects rather than patches for video inpainting and hence provides significant computational advantage by avoiding exhaustive search [43]. We use Dynamic Programming to holistically inpaint foreground objects with object templates that minimizes a sliding-window dissimilarity

cost function. This technique can effectively handle large regions of occlusions, inpaint objects that are completely missing for several frames, inpaint moving objects with complex motion, changing pose and perspective making it an effective alternative for video modification tasks in privacy protection applications. We will briefly describe our approach in Section 3.2 with more detailed analysis and performance analysis available in [43].

2.3 Privacy Data Management

A major shortcoming in most of the existing privacy protection systems is that once the modifications are done on the video for the purpose of privacy protection, the original video can no longer be retrieved. Consider a video surveillance network in a hospital. While perturbing or obfuscating the surveillance video may conceal the identity of patients, the process also destroys the authenticity of the signal. Even with the consensus from the protected patients, law enforcement and arbitrators will no longer have access to the original data for investigation. Thus, a privacy protection system must provide mechanism to enable users to selectively grant access to their private information. This is in fact the fundamental premise behind the Fair Information Practices [41, Ch.6]. In the near future, the use of cameras will become more prevalent. Dense pervasive camera networks are utilized not only for surveillance but also for other types of applications such as interactive virtual environment and immersive teleconferencing. Without jeopardizing the security of the organization, a flexible privacy data control system will become indispensable to handle complex privacy policy with large number of individuals to protect and different data requests to fulfil.

To tackle the management of privacy information, Lioudakis et.al recently introduce a framework which advocates the presence of a trusted middleware agent referred to as Discreet Box [16]. The Discreet Box acts as a three way mediator between the law, the users and the service providers. This centralized unit acts as a communication point between various parties and enforces the privacy regulations. Fidaleo et al. describe a secure sharing scheme in which the surveillance data is stored in a centralized server core [20]. A Privacy buffer zone, adjoining the central core, manages the access to this secure area by filtering appropriate personally identifiable information thereby protecting the data. Both approaches adopt a centralized management of privacy information making them vulnerable to concerted attacks. In contrast to these techniques, we propose a flexible software agent architecture that allows individual users to make the final decision on *every access* to their privacy data. This is reminiscent to a Data Right Management (DRM) system where the content owner can control the access of his/her content after proper payment is received [47]. Through a trusted mediator agent in our system, the user and the client agents can anonymously exchange data request, credential and authorization. We believe that our management system offers a much stronger form of privacy protection as the user no longer needs to trust, adhere or register his/her

privacy preferences with a server. Details of this architecture will be described in Section 3.1.

To address the issue of preserving the privacy information, the simplest solution is to store separately a copy of the original surveillance video. The presence of a separate copy becomes an easy target for illegal tempering and removal, making it very challenging to maintain the security and integrity of the entire system. An alternative approach is to scramble the privacy information in such a way that the scrambling process can be reversed using a secret key [5, 14]. There are a number of drawbacks of such a technique. First, similar to pixelation or blocking, scrambling is unable to fully protect the privacy of the objects. Second, it introduces artifacts that may affect the performance of subsequent image processing steps. Lastly, the coupling of scrambling and data preservation prevents other obfuscation schemes like object replacement or removal to be used.

On the other hand, we advocate the use of data hiding or steganography for preserving privacy information [48, 31, 34]. Using video data hiding, the privacy information is hidden in the compressed bit stream of the modified video and can be extracted when proper authorization can be established. The data hiding algorithm is completely independent from the modification process and as such, can be used with any modification technique. Data hiding has been used in various applications such as copyright protection, authentication, fingerprinting and error concealment. Each application imposes different set of constraints in terms of capacity, perceptibility and robustness [10]. Privacy data preservation certainly demands large embedding capacity as we are hiding an entire video bitstream in the modified video. As stated in Section 1, perceptual quality of the embedded video is also of great importance. Robustness refers to the survivability of the hidden data under various processing operations. While it is a key requirement for applications like copyright protection and authentication, it is of less concern to a well-managed video surveillance system targeted to serve a single organization. In Section 3.3, we describe a new approach of optimally placing hidden information in the Discrete Cosine Transform (DCT) domain that simultaneously minimizes both the perceptual distortion and output bitrate. Our scheme works for both high-capacity irreversible embedding with QIM [7] and histogram-based reversible embedding [6], which will be discussed in details as well.

3 Description of System and Algorithm Design

A high level description of our proposed system is shown in Figure 1. Green boxes are secured processing units within which raw privacy data or decryption keys are used. All the processing units are connected through an open local area network, and as such, all privacy information must be encrypted before transmission and the identities of all involved units must be validated. Red arrows show the flow of the compressed video and black arrows show the control information such as RFID data and key information.

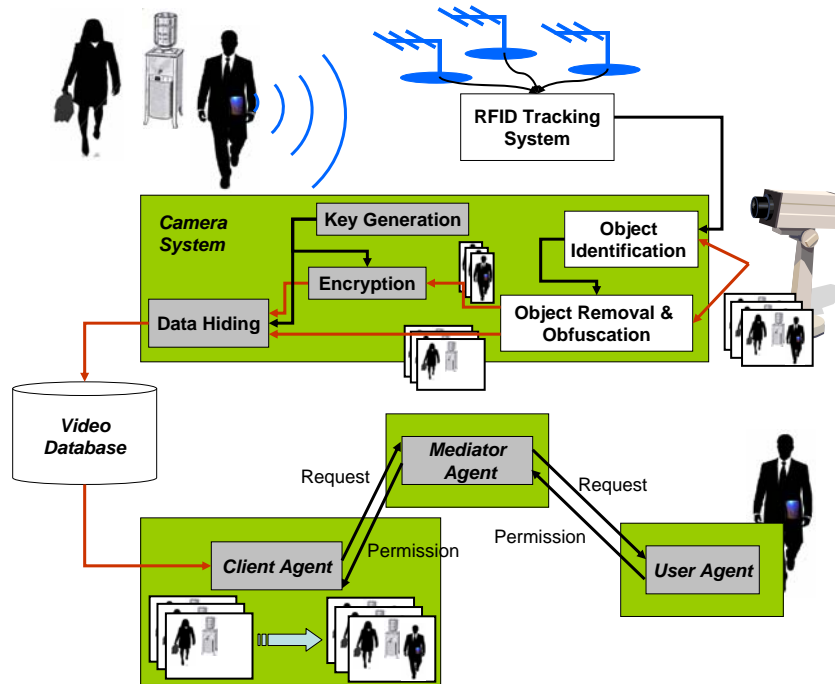


Fig. 1 High-level description of the proposed privacy-protecting video surveillance system.

Every trusted user in the environment carries an active RFID tag. The RFID System senses the presence of various active RFID tags broadcasting in different RF frequencies and triangulates them to compute their 2D coordinates in real time. It then consults the mapping between the tag ID and the user ID before creating an IP packet that contains the user ID, his/her 2D coordinates and the corresponding time-stamp. In order for the time-stamp to be meaningful to other systems, all units are synchronized using the Network Timing Protocol (NTP) [29]. NTP is an Internet Protocol for synchronizing multiple computers to within 10 ms, which is below the capturing period of both the RFID and the camera systems. To protect the information flow, the RFID system and all the camera systems are grouped into a IP multicast tree [33] with identities of systems authenticated and packets encrypted using IPsec [25]. The advantage of using IP multicast is that adding a new camera system amounts to subscribing to the multicast address of the RFID system. There is no need for the RFID system to keep track of the network status as the multicast protocol automatically handles the subscription and the routing of information. IPsec provides a transparent network layer support to authenticate each processing unit and to encrypt the IP packets in the open network.

In each camera system, surveillance video is first fed into the Object Identification and Tracking unit. The object tracking and segmentation algorithm used in the

camera system is based on our earlier work in [9, 43]. Background subtraction and shadow removal are first applied to extract foreground moving blobs from the video. Object segmentation is then performed during object occlusion using a real-time constant-velocity tracker followed by a maximum-likelihood segmentation based on color, texture, shape and motion. Once the segmentation is complete, we need to identify the persons with the RFID tags. The object identification unit visually tracks all moving objects in the scene and correlates them with the received RFID coordinates according to the prior joint calibration of the RFID system and cameras. This is accomplished via a simple homography that maps between the ground plane and the image plane of the camera. This homography translates the 2-D coordinates provided by the RFID system to the image coordinates of the junction point between the user and the ground plane. Our assumption here is that this junction point is visible at certain point during the entire object track, thus allowing us to discern the visual objects corresponding to the individuals carrying the RFID tags.

Image objects corresponding to individuals carrying the RFID tags are then extracted from the video, each padded with black background to make a rectangular frame and compressed using a H.263 encoder [22]. The compressed bitstreams are encrypted along with other auxiliary information later used by the privacy data management system. The empty regions left behind by the removal of objects are perceptually filled in the Video Inpainting Unit described in Section 3.2. The resulting protected video forms the cover work for hiding the encrypted compressed bitstreams using a perceptual-based rate-distortion optimized data hiding scheme described in Section 3.3. The data hiding scheme is combined with a H.263 encoder which produces a standard-compliant bitstream of the protected video to be stored in the database. The protected video can be accessed without any restriction as all the privacy information are encrypted and hidden in the bitstream. To retrieve these privacy information, we rely on the privacy data management system to relay request and permission among the client, the user and a trusted mediator software agent. In the following section, we provide the details of our privacy data management system.

3.1 Privacy Data Management

The goal of privacy data management is to allow individual users to control accessibility of their privacy data. This is reminiscent to a Data Right Management (DRM) system where the content owner can control the access of his/her content after proper payment is received. Our system is more streamlined than a typical DRM system as we have control over the entire data flow from production to consumption – for example, encrypted privacy information can be directly hidden in the protected video and no extra component is needed to manage privacy information. We use a combination of an asymmetric public-key cipher (1024-bit RSA) and a symmetric cipher (128-bit AES) to deliver a flexible and simple privacy data management system. RSA is used to provide flexible encryption of control and key information

while AES is computationally efficient for encrypting video data. Each user u and client c publish their public keys PK_u and PK_c while keeping the secret keys SK_u and SK_c to themselves. As a client has no way of knowing the presence of a user in a particular video, there is a special *mediator* m to assist the client in requesting permission from the user. The mediator also has a pair of public and secret keys PK_m and SK_m .

Suppose there are N users u_i with $i = 1, 2, \dots, N$ appeared in a video segment. We denote the protected video segment as V and the extracted video stream corresponding to user u_i as V_{u_i} . The Camera System prepares the following list of data to be embedded in V :

1. N AES-encrypted video streams $AES(V_{u_i}; K_i)$ for $i = 1, 2, \dots, N$, each using a randomly generated 128-bit key K_i .
2. An encrypted table of content $RSA(TOC; PK_m)$ using the mediator's public key PK_m . For each encrypted video stream V_{u_i} , the table of content TOC contains the following three data fields: a) the ID of user u_i ; b) the size of the encrypted bitstream; c) the RSA-encrypted AES key $RSA(K_i; PK_{u_i})$ using the public key of the user and d) other types of meta-information about the user in the scene such as the trajectory of the user or the specific events involved the user. Such information helps the mediator to identify the video streams that match the queries from client. On the other hand, this field can be empty if the privacy policy of the user forbids the release of such information.

The process of retrieving privacy information is illustrated in Figure 2. When a client wants to retrieve the privacy data from a video segment, the corresponding client agent retrieves the hidden data from the video and extracts the encrypted table of content. The client agent then sends the encrypted table of content and the specific query of interest to the mediator agent. Since the table of content is encrypted with the mediator's public key PK_m , the mediator agent can decrypt it using the corresponding secret key SK_m . However, the mediator cannot authorize the direct access to the video as it does not have the decryption key for any of the embedded video streams. The mediator agent must forward the request to those users that match the client's query for proper authorization. The request data packet for user u_j contains the encrypted AES key $RSA(K_j; PK_{u_j})$ and all the information about the requesting client c . If the user agent of u_j agrees with the request, it decrypts the AES key using its secret key SK_{u_j} and encrypts it using the client's public key PK_c before sending it back to the mediator. The mediator finally forwards all the encrypted keys back to the client which decrypts the corresponding video streams using the AES keys.

The above key distribution essentially implements a one-time pad for the encryption of each private video stream. As such, the decryption of one particular stream does not enable the client to decode any other video streams. The three-agent architecture allows the user to modify his/her private policy at will without first announcing it to everyone on the system. While the mediator agent is needed in every transaction, it contains no state information and thus can be replicated for load balancing. Furthermore, to prevent overloading the network, no video data is ever exchanged among agents. Finally, it is assumed that proper authentication is

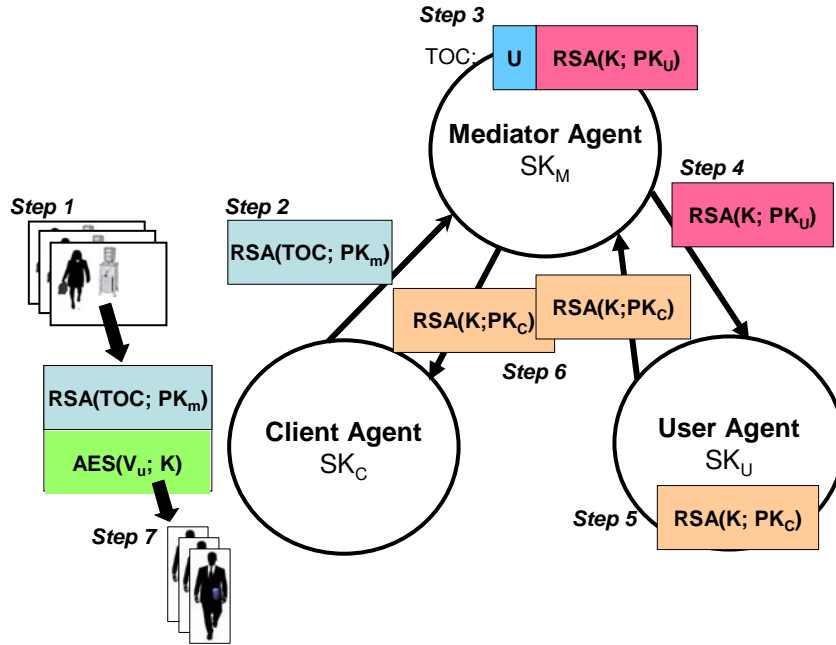


Fig. 2 Flow of privacy information: 1) Client extracts hidden data; 2) Encrypted TOC forwarded to Mediator; 3) Mediator decrypts TOC; 4) Mediator forwards encrypted video key to User; 5) User decrypts key and re-encrypts it with Client’s public key; 6) Encrypted video key forwarded to Client; 7) Client decrypts video stream depicting User.

performed for each transaction to authenticate the identity of each party and the integrity of the data.

3.2 Video Inpainting for Privacy Protection

In this section, we briefly describe the proposed video inpainting algorithm used in our Camera System. More details can be found in [43, 9]. Figure 3 shows the schematic diagram of the video inpainting process. The removal of the privacy object leaves behind an empty region or a hole in the video. The static portion of the hole is first filled by an adaptively updated background image if background information is available. Otherwise, image inpainting is performed based on the surrounding image statistics. The occluded moving foreground objects are inpainted by a two-stage process using the stored object templates.

In the first stage, we classify the frames in the hole as either partially-occluded or completely-occluded as shown in Figure 4. This is accomplished by comparing the size of the templates in the hole with the median size of templates in the database.

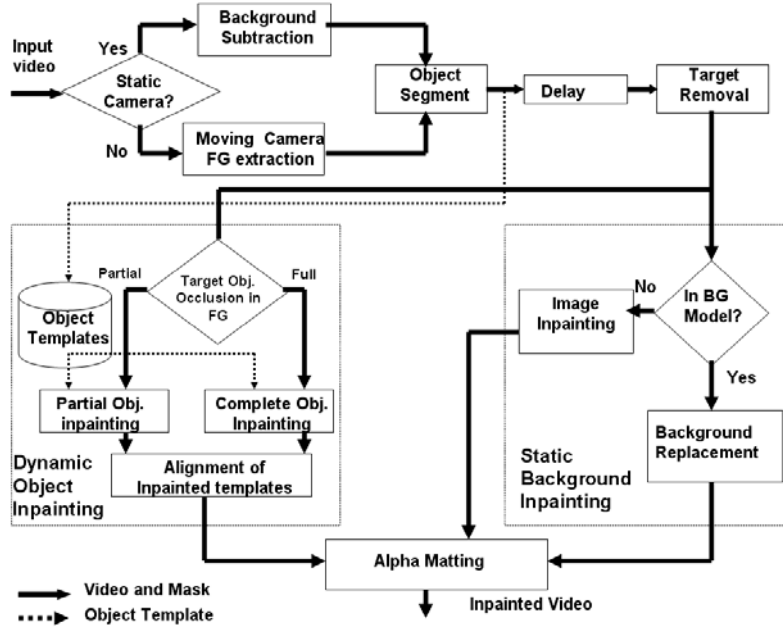


Fig. 3 Schematic diagram of the object removal and video inpainting system.

The reason of handling these two cases separately is that the availability of partially-occluded objects allow direct spatial registration with the stored templates, while completely-occluded objects must rely on registration done before entering and after exiting the hole.

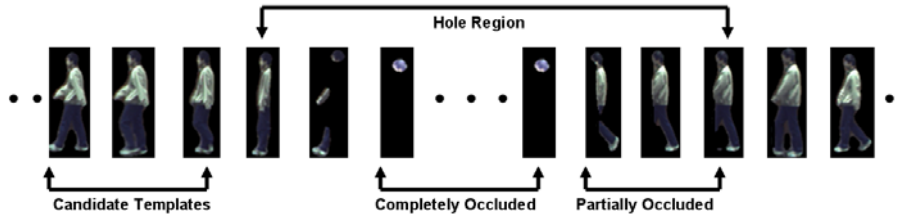


Fig. 4 Classification of the input frames into partially and completely occluded frames.

In the second stage, we perform a template search over the available object templates captured throughout the entire video segment. The partial objects are first completed with the appropriate object templates by minimizing a dissimilarity measure defined over a temporal window. Between a window of partially-occluded objects and a window of object templates from the database, we define the dissimilarity measure as the Sum of the Squared Differences (SSD) in their overlapping region plus a penalty based on the area of the non-overlapping region. The partially-

occluded frame is then inpainted by the object template that minimizes the window-based dissimilarity measure. Once the partially-occluded objects are inpainted, we are left with completely-occluded ones. They are inpainted by a Dynamic Programming (DP) based dissimilarity minimization process, but the matching cost is given by the dissimilarity between the available candidates in the database and the previously completed objects before and after the hole. The completed foreground and background regions are fused together using simple alpha matting. Figure 5 shows the result of applying our video inpainting algorithm to remove two people whose privacy needs to be protected.

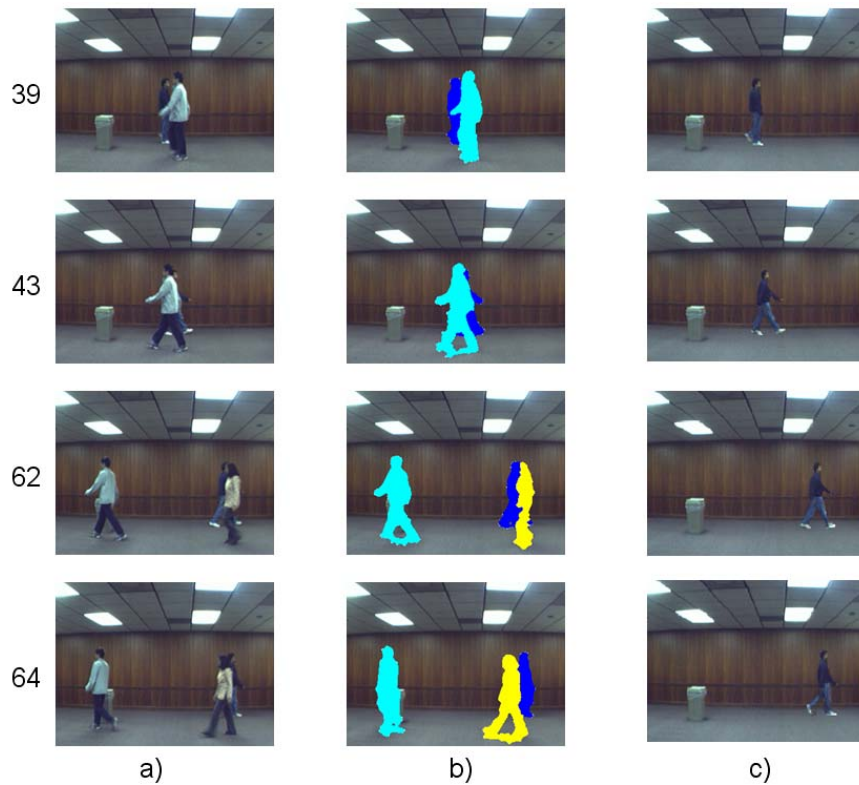


Fig. 5 a) The first column shows the original input sequence along with the frame number. b) The second column shows results of the tracking and foreground segmentation. c) The third column shows the inpainted result in which the individuals in the foreground are erased to protect their privacy. Notice that the moving person in the back is inpainted faithfully.

In many circumstances the trajectory of the person is not parallel to the camera plane. This can happen, for example, when we use ceiling mounted cameras or when the person is walking at an angle with respect to the camera position. Under this condition, the object undergoes a change in appearance as it moves towards or away from the camera. To handle such cases, we perform a normalization procedure

to rectify the foreground templates so that the motion trajectory is parallel to the camera plane. Under calibrated cameras, it is fairly straightforward to perform the metric rectification for normalizing the foreground volume. Otherwise, as explained in [43], we use features extracted from the moving person to compute the required geometrical constraints for metric rectification. After rectification, we perform our object-based video inpainting to complete the hole.

Our algorithm offers several advantages over existing state-of-the-art methods in the following aspects: First, using image objects allow us to handle large holes including cases where the occluded object is completely missing for several frames. Second, using object templates for inpainting provides significant speed up over existing patch-based schemes. Third, the use of a temporal window based matching scheme generates natural object movements inside the hole and provide smooth transitions at hole boundaries without resorting to any a prior motion model. Finally, our proposed scheme also provides a unified framework to address videos from both static and moving cameras and to handle moving objects with varying pose and changing perspective. We have tested the performance of our algorithm under varying conditions and the timing information for those sequences are given in Table 1. The results of the inpainting along with the original video sequences referred in the table are available in our project website at <http://vis.uky.edu/mialab/VideoInpainting.html>.

Table 1 Execution time on a Xeon 2.1Ghz machine with 4 Gigabyte of memory

Video	Segmentation	Inpainting	
		3-frame window	5-frame window
Three Person (Fig. 5)	30 secs	7.4 mins	10.2 mins
One Board	40 secs	3.5 mins	8.3 mins
Moving Camera	3 mins	2.6 mins	4.8 mins
Spinning Person	35 secs	12.6 mins	18.2 mins
Perspective	35 secs	3.6 mins	7.1 mins
Jumping Girl	30 secs	6.4 mins	11.4 mins

3.3 Rate Distortion Optimized Data Hiding Algorithm for Privacy Data Preservation

In this section, we describe a rate-distortion optimized data hiding algorithm to embed the encrypted compressed bitstreams of the privacy information in the inpainted video. Figure 6 shows the overall design and its interaction with the H.263 compression algorithm. Note that motion compensation is not used in the case of reversible embedding because the feedback loop in motion compensation will have to incorporate the hidden data in the residual frame, making the compensation process irreversible. Thus, we simply turn off the motion compensation for reversible embedding, resulting in a compression scheme similar to Motion JPEG (M-JPEG). The

embedding process is performed at frame level so that the decoder can reconstruct the privacy information as soon as the compressed bitstream of the same frame has arrived. Data is hidden only in the luminance DCT blocks which typically occupy the largest portion of the bit stream.

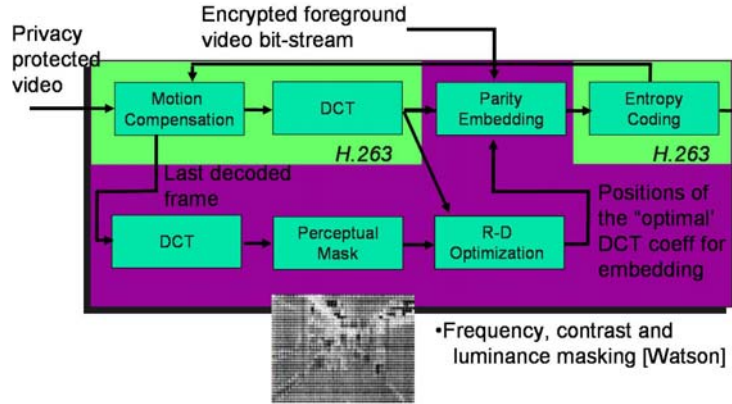


Fig. 6 Schematic diagram of the data hiding and video compression system

We first start with the irreversible data embedding where the modification to the cover video cannot be undone. Let $c(i, j, k)$ and $q(i, j, k)$ be the (i, j) -th coefficient of the k -th DCT block before and after quantization respectively. To embed a bit x into the (i, j, k) -th coefficient, we change $q(i, j, k)$ to $\tilde{q}(i, j, k)$ using the following embedding procedure:

1. If x is 0 and $q(i, j, k)$ is even, add or subtract one from $q(i, j, k)$ to make it odd. The decision of increment or decrement is chosen to minimize the difference between the reconstructed value and $c(i, j, k)$.
2. If x is 1 and $q(i, j, k)$ is odd, add or subtract one from $q(i, j, k)$ to make it even. The decision of increment or decrement is chosen to minimize the difference between the reconstructed value and $c(i, j, k)$.
3. $q(i, j, k)$ remains unchanged otherwise.

Following the above procedure, each DCT coefficient can embed at most one bit. Decoding can be accomplished using Equation (1):

$$x = (\tilde{q}(i, j, k) + 1) \bmod 2 \quad (1)$$

For the reversible embedding process, we exploit the fact that DCT coefficients follow a Laplacian distribution concentrated around zero with empty bins towards either ends of the distribution [6]. Due to the high data concentration at the zero bin, we can embed high-volume of hidden data at the zero coefficients by shifting the bins with values larger (or smaller) than zero to the right (or left). Let $L = \lceil M_k/Z \rceil$ where Z is the number of zero coefficients in this DCT block. We modify each DCT

coefficients $q(i, j, k)$ into $\tilde{q}(i, j, k)$ using the following procedure until all the M_k bits of privacy data are embedded.

1. If $q(i, j, k)$ is zero, extract L bits from the privacy data buffer and set $\tilde{q}(i, j, k) = q(i, j, k) + 2^{L-1} - V$ where V is the decimal value of these L privacy data bits.
2. If $q(i, j, k)$ is negative, no embedding is done in this coefficient and $\tilde{q}(i, j, k) = q(i, j, k) - 2^{L-1} - 1$.
3. If $q(i, j, k)$ is positive, no embedding is done in this coefficient but $\tilde{q}(i, j, k) = q(i, j, k) + 2^{L-1}$.

Similarly, at the decoder the level of embedding L is calculated first and then data extraction and distortion reversal is done using the following procedure.

1. If $-2^{L-1} < \tilde{q}(i, j, k) \leq 2^{L-1}$, L hidden bits can be obtained as the binary equivalent of the decimal number $2^{L-1} - \tilde{q}(i, j, k)$ and $q(i, j, k) = 0$.
2. If $\tilde{q}(i, j, k) \leq -2^{L-1}$, no hidden bit in this coefficient and $q(i, j, k) = \tilde{q}(i, j, k) + 2^{L-1} - 1$.
3. If $\tilde{q}(i, j, k) > 2^{L-1}$, no bit is hidden in this coefficient and $q(i, j, k) = \tilde{q}(i, j, k) - 2^{L-1}$.

Since only zero bins are actually used for data hiding, the embedding capacity is quite limited and hence it might be required to hide more than one bit at a coefficient in certain DCT blocks. Though the distortion due to this embedding is reversible at a frame level for an authorized decoder, the distortion induced is higher than the irreversible approach for a regular decoder.

To identify the embedding locations that cause the minimal disturbance to visual quality, we need a distortion metric in our optimization framework. Common distortion measures like mean square does not work for our goal of finding the optimal DCT coefficients to embed data bits: Given the number of bits to be embedded, the mean square distortion will always be the same regardless of which DCT coefficients are used as DCT is an orthogonal transform. Instead, we adopt the DCT perceptual model described in [10]. Considering the luminance and contrast masking of human visual system as described in [10], we calculate the final perceptual mask $s(i, j, k)$ that indicates the maximum permissible alteration to the (i, j) th coefficient of the k^{th} 8×8 DCT block of an image. With this perceptual mask, we can compute a perceptual distortion value for each DCT coefficient in the current frame as:

$$D(i, j, k) = \frac{QP}{s(i, j, k)} \quad (2)$$

where QP is the quantization parameter used for that coefficient.

In our joint data hiding and compression framework, we aim at minimizing the output bit rate R and the perceptual distortion D caused by embedding M bits into the DCT coefficients. By using a user-specified control parameter δ , we combine the rate and distortion into a single cost function as follows:

$$C = (1 - \delta) \cdot N_F \cdot D + \delta \cdot R \quad (3)$$

N_F is used to normalize the dynamic range of D and R . δ is selected based on the particular application which may favor the least amount of distortion by setting δ close to zero, or the least amount of bit rate increase by setting δ close to one. In order to avoid any overhead in communicating the embedding positions to the decoder, both of these approaches compute the optimal positions based on the previously decoded DCT frame so that the process can be repeated at the decoder.

The cost function in Equation 3 depends on which DCT coefficients used for the embedding. Thus, our optimization problem become

$$\min_{\Gamma} C(\Gamma) \text{ subjected to } M = N \quad (4)$$

where M is the variable that denotes the number of bits to be embedded, N is the target number of bits to be embedded, C is the cost function as described in Equation (3) and Γ is a possible selection of N DCT coefficients for embedding the data. Using Lagrangian Multiplier, this constrained optimization is equivalent to the following unconstrained optimization:

$$\min_{\Gamma} \Theta(\Gamma, \lambda) \text{ with } \Theta(\Gamma, \lambda) = C(\Gamma) + \lambda \cdot (M - N) \quad (5)$$

We can further simplify Equation (5) by decomposing it into the sum of similar quantities from each DCT block k :

$$\Theta(\Gamma, \lambda) = \sum_k C_k(\Gamma_k) + \lambda \cdot \left(\sum_k M_k - N \right) \quad (6)$$

To prepare for the above optimization, we need to first generate the curves between the cost and the number of embedded bits for all the DCT blocks. The cost function, as described in Equation (3) consists of both the distortion and the rate. The distortion is calculated using an L_4 norm pooling of distorted coefficients obtained from Equation (2). Rate increase is considerably more difficult as it depends on the run-length patterns. Embedding at the same coefficient may result in different rate increase depending on the order of embedding. While one can resort to dynamic programming techniques to compute the optimal curve, the computation time is prohibitive and we approximate the rate increase function using a greedy approach by embedding at the minimum cost position at each step. As the decoder does not know the actual bit to be embedded, the worst case scenario is assumed – both the distortion and the rate increase are computed by embedding the bit value at each step that leads to a bigger increase in cost. Once the cost curves are generated for all the DCT blocks, we can minimize the Lagrangian cost $\Theta(\Gamma, \lambda)$ for any fixed value of λ to find the distribution of embedding bits. λ value is chosen using the binary search such that the total bits over all the DCT blocks is just greater than or equal to the target embedding requirement. At this optimal slope, we get the number of bits to be embedded as the value of N which minimizes the unconstrained equation.

Figure 7 shows a sample frame of the hall monitor test sequence using irreversible embedding with different δ . The presence of hidden data is not visible

for both $\delta = 0$ and $\delta = 0.5$ and becomes marginally visible at $\delta = 1$. Table 2 shows



Fig. 7 234th frame of Hall Monitor Sequence after data hiding for $QP = 10$. Top Left: No Watermark ; Top Right: $\delta = 0$; Bottom Left: $\delta = 0.5$; Bottom Right: $\delta = 1$

the bit-rates for the hall monitor sequence at different δ values for both irreversible embedding using H.263 and reversible embedding using M-JPEG. The baseline for measurement is using separate files for storage – for H.263, the baseline is the sum of 119.15 kbps for the inpainted video and 81 kbps for the privacy information, making it a total of 200.15 kbps. For M-JPEG, the baseline is the sum of 2493 kbps for the inpainted video and 807 kbps for the privacy information or a total of 3300 kbps. The distortion is the pooled perceptual distortion measured based on the output from a standard compliant decoder with no knowledge of the data hidden inside. While the irreversible embedding causes less perceptual distortion than the reversible embedding, it has higher relative increase in bit-rate as well. The bit-rate increase also changes with QP. Fixing $\delta = 0.5$, table 3 shows the bit-rates of the inpainted video (R_o), the privacy information (R_p) and the inpainted video with privacy information embedded (R_e) at different QP's for the reversible embedding. Operating at higher quality (or lower QP) induces a lower relative increase in bit-rate, and irreversible embedding shows a similar trend as well.

Table 2 Bit rates and perceptual distortion of Hall Monitor Sequence (QP=10)

δ	Non Reversible Embedding)		Reversible Embedding	
	Rate Increase	Distortion	Rate Increase	Distortion
Separate files	0	0	0	0
0	63.8%	21.65	44.2%	127
0.5	50.9%	27	41.1%	135
1	43.4%	102	16.9%	255

Table 3 Bit-rates of reversible embedding at different QP's

QP	R_o	R_p	R_e	% increase
20	1560	710	3607	58.9
15	1885	744	3845	46.2
10	2493	807	4656	41.1
5	4018	960	6281	26.2

4 Challenges and opportunities in privacy protection

In this chapter, we describe a comprehensive solution of protecting privacy in a multi-camera video surveillance system. Selected individuals are identified with a real-time human tracking RFID system. The RFID system relays the information to each camera which tracks, segments, identifies and removes visual objects that correspond to individuals with RFID tags. To repair the remainder of the video, we employ an object-based video inpainting algorithm to fill in the empty regions and create the protected video. The original visual objects are encrypted and embedded into the compressed protected video using a rate-distortion optimal data hiding algorithm. Using a privacy data management system that comprises of three software agents, users can grant access to these privacy information to authenticated clients under a secure and anonymous setting.

Privacy protection in video surveillance is a new area with its requirements still heavily debated. It is a truly interdisciplinary topic that requires inputs from privacy advocates, legal experts, technologists and general public. The most pressing challenge facing the construction of such a system is the reliability of various components. RFID failures, less-than-perfect segmentation or inpainting even in a single video frame may expose the identity of a person and defeat the entire purpose of privacy protection. While our experiments indicate that these techniques perform well in controlled laboratory settings, their performances under different lighting conditions or in environments with reflective surfaces are questionable. While it might take years for these techniques to mature and be robust enough for privacy protection, it is possible to build a baseline system that simply blocks off all moving objects as a safety measure. As such, there is a spectrum of privacy protecting systems that tradeoff functionality with robustness and a careful study of their performance is an important step to move forward.

Acknowledgements The authors at University of Kentucky would like to acknowledge the support of Department of Justice under the grant 2004-IJ-CK-K055.

References

1. S. Avidan and B. Moshe. Blind vision. In *Proceedings of the 9th European Conference on Computer Vision*, pages 1–13, 2006.
2. A. M. Berger. *Privacy mode for acquisition cameras and camcorders*. Sony Corporation, us patent 6,067,399 edition, May 23 2000.
3. M. Bertalmio, A.L. Bertozzi, and G. Sapiro. Navier-stokes, fluid dynamics, and image and video inpainting. In *Proceedings of International Conference on Computer Vision and Pattern Recognition*, volume I, pages 355–362, Hawaii, 2001.
4. M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *Proceedings of ACM Conf. Comp. Graphics (SIGGRAPH)*, pages 417–424, New Orleans, USA, July 2000.
5. T. E. Boult. Pico: Privacy through invertible cryptographic obscuration. In *Proc. Computer Vision for Interactive and Intelligent Environments: The Dr. Bradley D. Carter Workshop Series*. University of Kentucky, 2005.
6. C. C. Chang, W. L. Tai, and M. H. Lin. A reversible data hiding scheme with modified side match vector quantization. In *Proceedings of the International Conference on Advanced Information Networking and Applications*, volume 1, pages 947–952, 2005.
7. Chen and Wornell. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. In *ISIT: Proceedings IEEE International Symposium on Information Theory*, 2000.
8. D. Chen, Y. Chang, R. Yan, and J. Yang. Tools for protecting the privacy of specific individuals in video. *EURASIP Journal on Advances in Signal Processing*, 2007:Article ID 75427, 9 pages, 2007. doi:10.1155/2007/75427.
9. S.-C. Cheung, J. Zhao, and V. Venkatesh M. Efficient object-based video inpainting. In *Proceedings of IEEE International Conference on Image Processing, ICIP 2006*, pages 705–708, 2006.
10. I.J. Cox, M.L. Miller, and J.A. Bloom. *Digital Watermarking*. Morgan Kaufmann Publishers, 2002.
11. A. Criminisi, Patrick Perez, and Kentaro Toyama. Region filling and object removal by exemplar-based inpainting. *IEEE Transactions on Image Processing*, 13(9):1200–1212, September 2004.
12. C. Diaz. *Anonymity and Privacy in Electronic Services*. PhD thesis, Katholieke Universiteit Leuven, 2005.
13. W. Diffie and S. Landau. *Privacy on the Line: The Politics of Wiretapping and Encryption*. The MIT Press, 1998.
14. Frdric Dufaux and Touradj Ebrahimi. Scrambling for video surveillance with privacy. *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, page 160, 2006.
15. Electronic Privacy Information Center, <http://www.epic.org/privacy/survey>. *Public Opinion on Privacy*, May 2006.
16. G. V. Lioudakis et al. A middleware architecture for privacy protection. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 51(16):4679–4696, November 2007.
17. L. Cranor et al. The platform for privacy preferences 1.0 (p3p1.0) specification. Technical report, World Wide Web Consortium (W3C), <http://www.w3.org/TR/P3P/>, 2002.
18. P. Jonathon Phillips et al. Frvt 2006 and ice 2006 large-scale results. Technical Report NISTRI 7408, National Institute of Standards and Technology, Marge 2007.
19. J. Wickramasuri et.al. Privacy protecting data collection in media spaces. *ACM Multimedia*, pages 48–55, October 2004.

20. D.-A. Fidaleo, H.-A. Nguyen, and M. Trivedi. The networked sensor tapestry (nest): a privacy enhanced software architecture for interactive analysis of data in video-sensor networks. In *VSSN '04: Proceedings of the ACM 2nd international workshop on Video surveillance & sensor networks*, pages 46–53, New York, NY, USA, 2004. ACM Press.
21. N. Hu and S.-C. Cheung. Secure image filtering. In *Proc. of IEEE International Conference on Image Processing (ICIP 2006)*, pages 1553–1556, Oct 2006.
22. ITU-T Recommendation H.263 Version 2. *Video Coding for Low Bitrate Communication Version 2*, 1998.
23. Jiaya Jia, Yu-Wing Tai, Tai-Pang Wu, and Chi-Keung Tang. Video repairing under variable illumination using cyclic motions. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 1:364–371, July 2006.
24. Y. T. Jia, S. M. Hu, and R. R. Martin. Video completion using tracking and fragment merging. In *Proceedings of Pacific Graphics*, volume 21, pages 601–610, 2005.
25. S. Kent and K. Seo. Security architecture for the internet protocol. Technical report, IETF RFC 4301, December 2005.
26. P. Kumar and A. Mittal. A multimodal audio visible and infrared surveillance system (maviss). In *International Conference of Intelligent Sensing and Information Processing*, 2005.
27. Yehuda Lindell and Benny Pinkas. Privacy preserving data mining. *Journal of Cryptology*, 15(3):177–206, 2002.
28. N. Megherbi, S. Ambellousi, O. Colot, and F. Cabestaing. Joint audio-video people tracking using belief theory. In *Proceedings of IEEE Conference on Advanced Video and Signal Based Surveillance*, 2005.
29. D. L. Mills. Network time protocol (version 3) specification, implementation and analysis. Technical report, IETF, 1992.
30. E. N. Newton, Latanya Sweeney, and B. Main. Preserving privacy by de-identifying face images. *IEEE transactions on Knowledge and Data Engineering*, 17(2):232–243, February 2005.
31. J. K. Paruchuri and S.-C. Cheung. Joint optimization of data hiding and video compression. In *To appear in IEEE International Symposium on Circuits and Systems (ISCAS 2008)*, 2008.
32. K. A. Patwardhan, G. Sapiro, and M. Bertalmio. Video inpainting under constrained camera motion. *IEEE Transactions On Image Processing*, 16(2):545–553, Feb 2007.
33. B. Quinn and K. Almeroth. Ip multicast applications: Challenges and solutions. Technical report, IETF RFC 3170, September 2001.
34. J. Paruchuri S.-C. Cheung and T. Nguyen. Managing privacy information in pervasive camera networks. In *Proceedings of IEEE International Conference on Image Processing, ICIP 2008*, 2008.
35. H. Schantz. Near field phase behavior. In *Proceedings of IEEE APS Conference*, 2005.
36. H. Schantz and R. Depierre. System and method for near-field electromagnetic ranging. Technical Report 6,963,301, U.S. Patent, 2005.
37. J. Schiff, M. Meingast, D. Mulligan, S. Sastry, and K. Goldberg. Respectful cameras: Detecting visual markers in real-time to address privacy concerns. In *International Conference on Intelligent Robots and Systems (IROS)*, 2007.
38. A. Senior, S. Pankanti, A. Hampapur, Y.-L. Tian L. Brown, and A. Ekin. Blinkering surveillance: Enabling video privacy through computer vision. *Security and Privacy*, 3:50–57, 2005.
39. E. Shakshuki and Y. Wang. Using agent-based approach to tracking moving objects. In *Proceedings of 17th International Conference on Advance Information Networking and Application*, 2003.
40. Takaaki Shiratori, Yasuyuki Matsushita, Sing Bing Kang, and Xiaoou Tang. Video completion by motion field transfer. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 411–418, June 2006.
41. D. J. Solove. *The Digital Person: Technology and Privacy in the Information Age*. New York University Press, 2004.
42. SPAWAR System Center. *Correctional Officer Duress Systems: Selection Guide*, November 2003.

43. M. Vijay Venkatesh, S.-C. Cheung, and J. Zhao. Efficient object-based video inpainting. *Pattern Recognition Letters : Special issue on Video-based Object and Event Analysis*, 2008.
44. H. Wactlar, S. Stevens, and T. Ng. *Enabling Personal Privacy Protection Preferences in Collaborative Video Observation*. NSF Award Abstract 0534625, <http://www.nsf.gov/awardsearch/showAward.do?awardNumber=0534625>.
45. J. Wada, K. Kaiyama, K. Ikoma, and H. Kogane. *Monitor camera system and method of displaying picture from monitor camera thereof*. Matsushita Electric Industrial Co. Ltd., european patent, ep 1 081 955 a2 edition, April 2001.
46. Yonatan Wexler, Eli Shechtman, and Michal Irani. Space-time completion of video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):463–476, 2007.
47. W. Zeng, H. Yu, and C.-Y. Lin. *Multimedia Security Technologies for Digital Rights Management*. Academic Press, 2006.
48. W. Zhang, S.-C. Cheung, and M. Chen. Hiding privacy information in video surveillance system. In *Proceedings of the 12th IEEE International Conference on Image Processing*, Genova, Italy, September 2005.
49. Yunjun Zhang, Jiangjian Xiao, and Mubarak Shah. Motion layer based object removal in videos. In *Proceedings of the Seventh IEEE Workshops on Application of Computer Vision*, volume 1, pages 516–521, 2005.
50. J. Zhao and S.-C. Cheung. Multi-camera surveillance with visual tagging and generic camera placement. In *Proceedings of ACM/IEEE International Conference on Distributed Smart Cameras*, Sept. 2007.