Region-of-Interest Scrambling for Scalable Surveillance Video using JPEG XR

Hosik Sohn
Image and Video Systems Lab
KAIST
Daejeon, Korea
sohnhosik@kaist.ac.kr

Wesley De Neve
Image and Video Systems Lab
KAIST
Daejeon, Korea
wesley.deneve@kaist.ac.kr

Yong Man Ro
Image and Video Systems Lab
KAIST
Daejeon, Korea
ymro@ee.kaist.ac.kr

ABSTRACT

Present-day video surveillance systems are often required not to intrude upon the privacy of the general public. In this paper, we discuss a privacy-protected video surveillance system that makes use of the JPEG XR standard. This standard offers a low-complexity solution for the scalable coding of high-resolution images. To address privacy concerns, face regions are detected and subsequently scrambled in the transform domain, taking into account the spatial and quality scalability features of JPEG XR. A number of experiments were conducted in order to investigate the efficiency of our video surveillance system, considering bit stream overhead and security aspects.

Categories and Subject Descriptors

I.4.9 [**Applications**]

General Terms

Experimentation, Performance, Security

Keywords

Image coding, JPEG XR, Scrambling, Video surveillance

1. INTRODUCTION

Nowadays, video surveillance systems are omnipresent in public places. These systems are often characterized by high-speed network connections, plenty of storage capacity, and a high computational power. Moreover, thanks to continuously improving computer vision algorithms, video surveillance systems are increasingly able to analyze and understand events of interest.

In video surveillance systems, spatial resolution and visual quality are critical factors for the performance of computer vision algorithms. Indeed, the use of high-resolution and high-quality video content improves the overall performance of computer vision algorithms targeting object detection, identification, and tracking. A high spatial resolution and a high visual quality are also important for legal reasons. For example, in the UK, the minimum resolution of traffic Closed-Circuit Television (CCTV) cameras, used for the detection of unlawful drivers, was recently the topic of a legal debate [1]. In addition, privacy concerns are also on the rise. People are being monitored without having given their consent or without having knowledge about these activities. The increasing use of high-resolution surveillance cameras will even pose more threats to the privacy of individuals.

In this paper, we propose a video surveillance system using the state-of-the-art JPEG Extended Range (JPEG XR) format for scalable image coding [2]. This coding format comes with a low computational complexity, while offering a high image quality and a high flexibility of use in diverse usage environments. In particular, JPEG XR can be seen as a low-complexity alternative to JPEG 2000, which is frequently used in current video surveillance systems. JPEG XR is expected to be ratified as an international standard in the course of this year (formally denoted as ISO/IEC 29199-2). Further, for the purpose of privacy protection, our system detects and protects face regions, which are considered privacy sensitive. Protection is realized using different scrambling techniques operating in the transform domain, taking into account the scalability provisions of JPEG XR.

The rest of this paper is organized as follows. In Section 2, we briefly discuss the basic coding tools of JPEG XR, while our scrambling approach is outlined in Section 3. Several experimental results are presented in Section 4 to demonstrate the efficiency of our approach. Finally, Section 5 provides a number of concluding remarks.

2. IMAGE CODING USING JPEG XR

This section briefly reviews the coding tools of JPEG XR that enable scalability and region-of-interest (ROI) coding. For a more detailed discussion, we would like to refer the reader to [3].

2.1 Scalable Intra Coding

In our surveillance system, each video frame is intra-coded using JPEG XR. The main technical benefit of using JPEG XR as an intra video codec can be found in its low computational complexity, while offering image quality and scalability provisions that are, from a practical point of view, similar to that of Motion JPEG 2000 and the Scalable High Intra Profile of H.264/AVC Scalable Video Coding (SVC). This observation holds especially true for the JPEG 2000 standard. For example, in JPEG 2000, the wavelet transform is used as a global transform, operating at the level of a tile, and typically requiring more memory bandwidth than the 4×4 block-based transforms of SVC and JPEG XR [4].

The transform used by JPEG XR is denoted as a two-staged hierarchical Lapped Biorthogonal Transform (LBT). In frequency

MM'09, October 19–24, 2009, Beijing, China.
Copyright 2009 ACM 978-1-60558-608-3/09/10...$10.00.

mode, a JPEG XR bit stream offers support for both spatial and quality scalability, thanks to a partitioning of the transform coefficients of a particular tile into four subbands: the DC subband (containing a single second-stage DC coefficient for each macroblock or MB in the tile), the low pass or LP subband (containing 15 second-stage transform coefficients for each MB), the high pass or HP subband (containing the significant part of the 240 first-stage transform coefficients of each MB), and the Flexbits subband (containing the refinement bits of the 240 first-stage transform coefficients of each MB). Significant bits and refinement bits are also computed for the DC and LP coefficients. However, in contrast to the 240 first-stage transform coefficients, the significant bits and refinement bits of DC and LP coefficients are not stored in a separate subband.

A trivial form of quality scalability can be realized by removing the Flexbits subbands, or part thereof, from a JPEG XR bit stream. Spatial scalability is supported by additionally removing the HP and LP subbands, each time resulting in a reduction of the spatial resolution by a factor of four along the horizontal and vertical axis. Intermediate resolutions can be achieved by relying on client-side down sampling techniques, enabling a complexity trade-off between the server, the bit stream extractor, and the client. Dependent on the application targeted, spatial scalability in JPEG XR can also be seen as coarse-grained quality scalability when still displaying the adapted image at its original resolution.

2.2 ROI Representation

In JPEG XR, an image may consist of several spatial tiles. Each spatial tile represents a group of spatially adjacent macroblocks. Since there is no coding dependency between spatially adjacent tiles (except when the overlap transform is enabled), tiles can be used to represent an ROI. As such, ROI extraction in JPEG XR can be realized by extracting spatial tiles in the compressed domain, in both spatial and frequency mode. This feature of JPEG XR is also known as fast tile extraction.

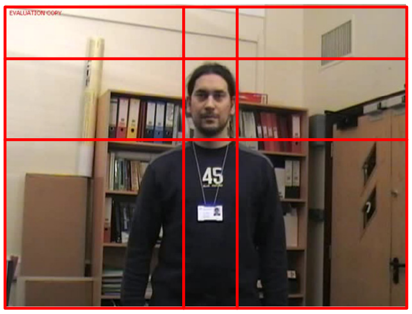


Figure 1. ROI representation in JPEG XR.

Two types of tile layouts are possible: a uniform and a non-uniform tile grid. In the uniform tile layout, each tile has the same width and height, while the non-uniform layout permits the use of tiles with different widths and heights (tiles on the same row still need to have the same height, while tiles on the same column still need to have the same width). The non-uniform tile layout is illustrated in Figure 1. Note that the use of a fine-grained tile grid may significantly decrease the coding efficiency [5]. This will also be discussed in Section 4 for our use case.

3. SCRAMBLING

In this section, we outline our scrambling approach for protecting privacy-sensitive face regions in surveillance video content. We assume the location of a face region is known before encoding.

3.1 Proposed Encoder Architecture

Figure 2 illustrates the architecture of our modified JPEG XR encoder. In particular, before entropy coding, scrambling is applied to the transform coefficients in the DC and LP subbands. The information stored in the HP and Flexbits subbands is not altered due to its limited impact on the visual quality. This will be explained in more detail in Section 3.2.

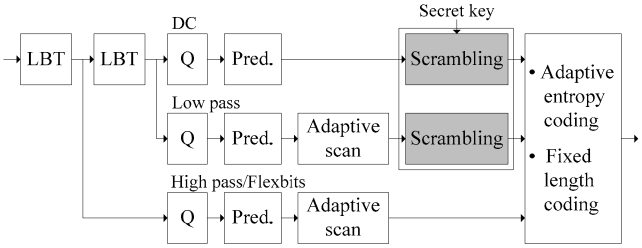


Figure 2. Architecture of our modified JPEG XR encoder.

Note that a secret key is used as a seed value to generate pseudo-random numbers. Therefore, only an authorized user (i.e., a user who knows the secret key) is able to invert the scrambling process during decoding. A detailed description of the actual bit stream extraction, decoding, descrambling, and key management processes [6] is omitted due to space limitations.

3.2 Subband-Adaptive Scrambling

As shown in Figure 2, a subband-adaptive approach is followed in order to scramble privacy-sensitive face regions. This approach is motivated by the following observation: when scrambling a particular subband, a trade-off exists between the visual importance of the subband (the information in the DC subbands is for instance visually more important than the information in the LP subbands), the available amount of coded data in the subband (the number of coefficients increases when going from the DC subband to the HP subband, and hence the amount of compressed data), the level of security offered by the scrambling technique used, the effect on the coding efficiency, and the computational complexity of the scrambling technique used.

3.2.1 Scrambling for DC Subbands

In a DC subband, a limited amount of data is available for the purpose of scrambling. Indeed, each macroblock in a tile only contributes a single DC coefficient to the DC subband of that particular tile. Therefore, we propose to apply scrambling at the level of individual bits in order to ensure a sufficient level of protection. Specifically, we propose to apply both Random Sign Inversion (RSI; [7]) and Random Bit Flipping (RBF) to DC subbands. RSI pseudo-randomly flips the sign of DC coefficients as follows:

$$D^e = \begin{cases} -D, if\ r = 1 \\ D, otherwise \end{cases}. \qquad (1)$$

where D denotes the data to be scrambled and where D^e denotes the pseudo-randomly sign-flipped data. As the sign of DC coefficients is signaled using a simple Boolean flag in JPEG XR, the use of RSI does not affect the coding efficiency. RBF flips bits by applying an XOR operation between input bits and bits belonging to a pseudo-random data stream:

$$b_i^e = b_i \; XOR \; r, \; where \; b_i \in B, b_i^e \in B^e, \; and \; r \in R. \qquad (2)$$

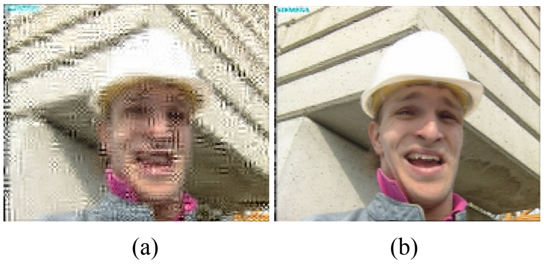
In Equation 2, B denotes the data to be encrypted while B^e denotes the encrypted data. Further, b_i denotes the i^{th} bit of B and R denotes the set of pseudo-random bits. In JPEG XR, each DC coefficient is partitioned into a significant part and a remainder part (i.e., DC refinement bits). The significant part is again partitioned into a level value and level refinement bits. The level value is signaled using variable length codes, while both DC refinement bits and level refinement bits are signaled using fixed length codes. RBF is only applied to the DC refinement bits and the level refinement bits. By combining the low-complexity RSI and RBF scrambling techniques, the coefficients in a DC subband can be significantly altered without affecting the coding efficiency, which is an important characteristic for mobile devices.

3.2.2 *Scrambling for LP Subbands*

An LP subband is visually less important than a DC subband, but visually more important than an HP subband. Also, an LP subband contains more transform coefficients than a DC subband, but less transform coefficients than an HP subband. Therefore, we propose to apply Random Permutation (RP; [7]) to the transform coefficients in an LP subband. RP offers a higher level of protection than RSI or RBF as RP allows for a higher number of possible combinations. However, RP comes with a decrease in coding efficiency since this scrambling technique breaks entropy coding. In our experiments, we observed that the decrease in coding efficiency was limited (less than 6.6% for a worst case scenario). This will be discussed in more detail in Section 4.1.

3.2.3 *HP and Flexbits Subbands*

In our experiments, we have observed that the visual impact of a scrambled HP subband lowers when the resolution increases (as the content of the HP subband represents high frequency information). Figure 3 shows two images taken from "Foreman". The HP subband is scrambled using RP, once at QCIF and once at 4CIF resolution (Flexbits not shown).



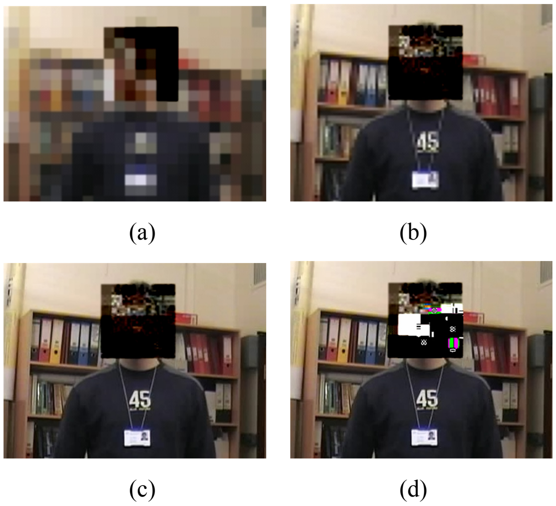
(a) (b)

Figure 3. Visual impact of a scrambled HP subband: (a) QCIF resolution and (b) 4CIF resolution.

As shown in Figure 3(b), the visual effect of a scrambled HP subband can hardly be seen at 4CIF resolution. The QCIF image in Figure 3(a) even demonstrates that a face region with a sufficiently high resolution cannot be concealed adequately at a low spatial resolution. Further, we have also observed that the application of RP to an HP subband significantly lowers the coding efficiency (in the order of 24% to 52%). When also taking into account that scrambled DC and LP subbands already alter the visual quality significantly, and the fact that the application of RP to HP subbands also requires a significant number of additional computations (as HP subbands contain significantly more compressed data than the DC and LP subbands), we propose not to scramble HP subbands. Following a similar reasoning, we also propose not to scramble Flexbits subbands.

4. EXPERIMENTAL RESULTS

We have implemented the proposed scrambling approach in the JPEG XR encoder available in the HD Photo Device Porting Kit (DPK) 1.0 provided by Microsoft [8]. The video sequence used in our experiment is "montinas_toni". This video sequence, part of the Surveillance Performance EValuation Initiative (SPEVI) dataset [9], has VGA resolution and a frame rate of 25 frames per second. The first eight seconds of the video sequence were used in our experiment. Further, the average size of the face region in the video sequence is 6x6 macroblocks.



(a) (b)

(c) (d)

Figure 4. Privacy-protected surveillance video: (a) DC, (b) DC + LP, (c) DC + LP + HP, and (d) DC + LP + HP + Flexbits.

Figure 4 shows the visual effect of our scrambling approach for "montinas_toni", varying the number of decoded subbands (cropped for visualization purposes).

4.1 Bit Stream Overhead Analysis

Table 1 shows the bit stream overhead according to the tile size, varying the bit rate. For each bit rate, the overhead is computed using as reference an image coded in spatial mode with no tiles. The second column of Table 1 represents the overhead when using the non-uniform tile layout as shown in Figure 1, while the other columns represent the overhead when using a uniform tile layout. For example, the label 1x1 MB refers to a uniform tile layout consisting of 40x30 tiles in VGA resolution. As shown in Table 1, the combined use of a small tile size and a uniform tile layout may significantly decrease the coding efficiency. This can be attributed to a broken entropy coding, an increasing number of tile headers, and an increasing number of entries in the index table. Also, for a particular tile configuration, the overhead becomes higher as the bandwidth decreases (the same syntax structures are used to signal a lower amount of compressed image data).

Table 1. Bit stream overhead according to the tile size

Tile grid Bit rate (Kbit/s)	9 tiles (%)	1×1 MB (%)	5×5 MB (%)	10×10 MB (%)
629	10.6	771.9	72.2	16.5
955	7.3	482.1	47.6	11.2
1348	4.5	323.0	32.8	7.4
1964	2.8	207.9	21.5	4.6
2809	1.9	135.5	14.2	3.2
4004	1.2	86.8	8.9	1.9
5781	0.5	54.4	5.0	0.6
8158	0.2	35.0	3.0	0.2

Figure 5 shows the average bit stream overhead, caused by the proposed scrambling approach. The overhead is shown for two cases: scrambling of the whole image and scrambling of the ROI (both images are coded using the non-uniform tile layout; the overhead caused by the use of tiles is not taken into account).

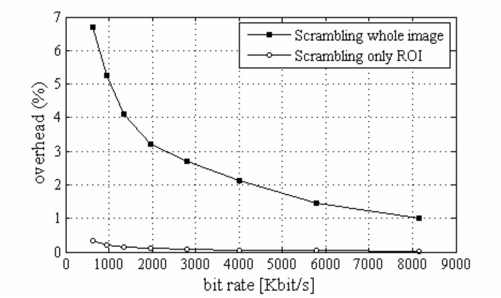


Figure 5. Bit stream overhead introduced by scrambling.

As shown in Figure 5, in the worst case (i.e., at a bit rate of 629 Kbit/s), the overhead is approximately 6.6% when scrambling the whole image and about 0.34% when only scrambling the ROI.

4.2 Security Considerations

This section analyzes the level of protection offered by the proposed scrambling technique against a brute force attack. For one macroblock, the combined application of RSI and RBF at the level of the DC coefficient results in 2^{N+1} possible combinations (N denotes the number of bits used to represent the fixed length part of the DC coefficient), while the application of RP to the LP coefficients results in a total of 15! possible combinations. Figure 6 shows the average number of bits assigned to the fixed length part of the DC coefficient. As such, the total number of combinations required to break the protection of a macroblock is equal to (2^{N+1} + 15!).

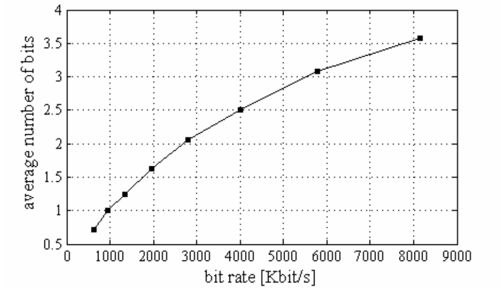


Figure 6. Average number of bits used to represent the fixed length part of a DC coefficient.

The compressed video bit stream at 629Kbit/s has the lowest level of protection. A brute force attack at the level of a single macroblock requires evaluating ($2^{1.72}$+15!) combinations. Since the size of the ROI is equal to 6×6 macroblocks, a brute force attack at the level of the ROI requires evaluating $(2^{1.72})^{36} + (15!)^{36}$ combinations: $(2^{1.72})^{36}$ evaluations are required for the DC subband and $(15!)^{36}$ evaluations for the LP subband. As decoding and descrambling of the DC subband requires about 1.9 ms on a quad-core 2.0 GHz processor, the time needed to generate all possible face regions is approximately equal to 2.3×10^{12} hours. This number shows that the proposed scrambling approach provides a feasible level of protection against a brute force attack (on the condition that the size of the ROI is sufficiently large).

5. CONCLUSIONS

This paper discussed an approach for scrambling privacy-sensitive face regions in scalable surveillance video coded using JPEG XR. Our approach is the result of a trade-off between the visual importance of subbands, the amount of coded data in the subbands, the level of security offered by a particular scrambling technique, the effect of scrambling on the coding efficiency, and the computational complexity of the scrambling technique used. The results show that privacy-sensitive regions can be successfully concealed with a feasible level of protection.

Future research will focus on conducting comparative experiments with Motion JPEG 2000 and the Scalable High Intra Profile of SVC.

6. REFERENCES

[1] The Guardian, "Ahead of G20 summit, council told to switch off illegal £15m CCTV network," Available on: http://www.guardian.co.uk/uk/2009/mar/30/cctv-london-government-transport-g20.

[2] Srinivasan, S., Tu, C., Zhou, Z., Ray, D., Regunathan, S., and Sullivan, G. J. An Introduction to the HD Photo Technical Design. JPEG document WG1 N4183.

[3] Srinivasan, S., Tu, C., Regunathan, S. L., and Sullivan, G. J. HD Photo: A new image coding technology for digital photography. Proc. of SPIE, vol. 6696, pp. 66960A, August 2007.

[4] Tran, T. D., Liu, L., and Topiwala, P. Performance comparison of leading image codecs: H.264/AVC Intra, JPEG2000, and Microsoft HD Photo. Proc. of SPIE, Vol. 6696, pp. 66960B, October 2007.

[5] Perra, C. and Giusto, D. An image browsing application based on JPEG XR. International Workshop on Content-Based Multimedia Indexing (CBMI). pp. 396-401, June 2008.

[6] Won, Y. G., Bae, T. M., and Ro, Y. M. Scalable protection and access control in full scalable video coding. Lecture Notes in Computer Science (LNCS), 4283, pp. 407-421, November 2006.

[7] Zeng, W. and Lei, S. Efficient frequency domain video scrambling for content access control. ACM Multimedia, pp. 285-294, 1999.

[8] HD Photo Device Porting Kit 1.0, Available on: http://www.microsoft.com/whdc/xps/hdphotodpk.mspx.

[9] Surveillance Performance EValuation Initiative (SPEVI) Datasets. Available on: http://www.elec.qmul.ac.uk/staffinfo/andrea/spevi.html.