Mosaic-based Privacy-protection with Reversible Watermarking

Yuichi Kusama, Hyunho Kang and Keiichi Iwamura
Department of Electrical Engineering, Tokyo University of Science,
6-3-1 Niijuku, Katsushika-ku, Tokyo 125-8585, Japan

Keywords: Video Surveillance, Privacy Protection, Reversible Watermarking.

Abstract: Video surveillance has been applied to many fields, specifically for detecting suspicious activity in public places such as shopping malls. As the use of video-surveillance cameras increases, so too does the threat to individual privacy. Therefore, video-surveillance technologies that protect individual privacy must be implemented. In this study, we propose a scheme in an MPEG2 video-encoding environment that successfully employs mosaicking, encryption, and restoration of faces captured in videos.

1 INTRODUCTION

Surveillance cameras are installed in various places, such as street corners, convenience stores, and metro stations. The main purpose of installing surveillance cameras is to deter criminals and record criminal activity. However, privacy is an issue with surveillance cameras, particularly because it is unclear how to handle ostensibly private information such as facial identifications. Thus, we may display surveillance-camera pictures on television or in newspapers or Internet articles, but privacy is typically protected by applying a mosaic to the faces of bystanders.

In this way, it is often necessary to conceal faces such that individuals are not identified. Such masking is typically accomplished with a mosaic (or 'pixelization') applied to privacy-infringing areas of the surveillance-camera picture. However, it is sometimes desirable for mosaicked areas of the picture to be restored, if the video is used to investigate some crime for instance. Therefore, when utilizing a surveillance-camera picture for some legitimate reason, techniques must be available to restore concealed faces.

Techniques to conceal private areas are common. Conventional mosaic techniques can divided into reversible and irreversible conversions. Irreversible conversions take the mean or the median of the target range and change the target range to the mean or the median. On the other hand, for reversible conversions, it is common to change the target-range pixel location. However, even if a reversible mosaic is applied beforehand, individuals can nevertheless be identified, because reversible conversions merely change the pixel location of the target range. Therefore, in this paper, we suggest a novel and reversible mosaic technique that encrypts the image.

Our proposed method encrypts the information needed to remove a mosaic, and it embeds the encrypted information using reversible watermarking when a mosaic is applied. This method ensures privacy protection, insofar as only valid users who know the encryption key can restore a mosaic. Moreover, upon reversing a mosaic, the image is restored without any deteriorated information.

Watermarking is a technique to embed information in a way that cannot be perceived by the user. Watermarking can be classified into reversible watermarking and irreversible watermarking. With reversible watermarking, the content is identical to the original image when the watermark is removed. Therefore, reversible watermarking is used for medical imaging, for instance, where the deterioration of content is unacceptable. Irreversible watermarking, however, cannot reverse the watermark, even after the information is extracted.

There have been several methods proposed to address the issue of privacy in surveillance cameras (Dufaux and Ebrahimi, 2008), (Carrillo et al., 2009), (Peng et al., 2013), (Li et al., 2009), (Yu and Babaguchi, 2007), (Saini et al., 2014). In this paper, we propose a new method for protecting privacy, using reversible watermarking to encrypt information and a novel mosaic technique.

We implemented the proposed method in order to meet the following three conditions:

Kusama Y., Kang H. and Iwamura K..
Mosaic-based Privacy-protection with Reversible Watermarking.
DOI: 10.5220/0005562500980103
In *Proceedings of the 12th International Conference on Signal Processing and Multimedia Applications* (SIGMAP-2015), pages 98-103
ISBN: 978-989-758-118-2

(1) The picture that is masked by a mosaic is natural (seamless) and retains as a digital watermark the information needed to restore the picture.

(2) The embedded information is preserved even after the video is compressed.

(3) The illegal restoration of the embedded information is prohibited, and only an authorized person can remove a mosaic.

This paper organized as follows. Section 2 explains the background research. In Section 3, we explain proposed method. Section 4 presents the experimental results from a simulation, and Section 5 concludes the paper.

2 PRELIMINARY

2.1 MPEG2

MPEG2 is a method to compress digital videos. Once compressed, the original quality cannot be restored.

2.1.1 Picture Types in MPEG2

With MPEG2 compression, there are three types of pictures defined: I, P and B.

An example of the MPEG2 frame constitution is shown in Figure 1.

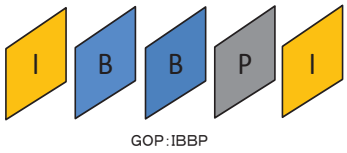


Figure 1: Example of the MPEG2 frame constitution.

I-frames: Encoded with a frame of its own, without using information from other frames.

P-frames: Encoded using the forward-motion-compensated prediction from the preceding I or P frame.

B-frames: Encoded using bidirectional-motion-compensated prediction from previous and subsequent I or P frames.

In addition, with MPEG2, the frames are not independent, and compression is performed by a unit of the frame called the GOP (group of pictures). In this study, the GOP is composed exclusively of I-frames.

2.1.2 MPEG2 Video Encoding

A block diagram for MPEG2 video encoding is shown in Figure 2.

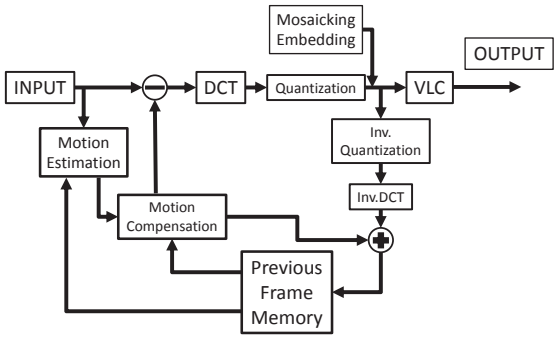


Figure 2: Example of the MPEG2 encoder.

The input image becomes the bit-stream output after 2D-DCT (two-dimensional discrete cosine transform) processing, quantization, and VLC (variable-length coding). When a frame must refer to other frames, such as P-frames for forward prediction, we perform inverse quantization and inverse 2D-DCT processing to the preceding frame. After temporarily restoring a forward frame and converting a predicted frame to a local decoder, the frame is saved in the Previous Frame Memory. Next, we compare the previous frame with the current frame using motion estimation and calculate a motion vector. We then generate a motion-compensated frame from the motion vector along with a forward frame . The difference between the motion-compensated frame and current frame is calculated to determine the prediction errors. Finally, we generate a P-frame applying the prediction errors to 2D-DCT, quantization, and VLC. In our study, we embed information in watermarks and apply a mosaic for privacy protection.

2.1.3 MPEG2 Video Decoding

A block diagram for MPEG2 video decoding is shown in Figure 3.

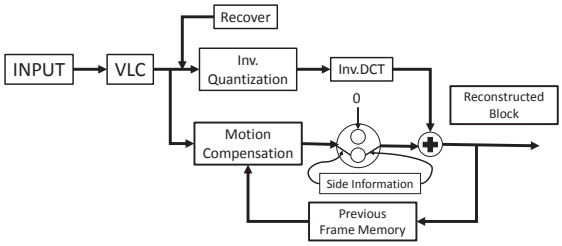


Figure 3: Example of the MPEG2 decoder.

The input bit-stream encoded with the MPEG2 encoder is decoded by VLC decoding, inverse quantization, and inverse 2D-DCT processing . In addition, the motion compensation used for encoding is selected among the encoding information from the

encoded area, and a reference signal is obtained after motion compensation. We generate the decoded frame by adding the prediction errors formed in the reference signal and encoding. In our study, we restore mosaicked area by adding operations just before inverse quantization.

2.2 Reversible Watermarking

We employ (Xuan et al., 2007) as a method to embed information for restoring mosaicked areas.

This approach applies JPEG compression to the DCT coefficient after quantization, and it embeds reversible information in the JPEG images. The quality of the picture after embedded this information remains high, and this was the main reason for employing this method. In our study, we applied MPEG-2 compression. Because we secured an embedding domain, we must change the range in order to embed the mosaic information, as shown in Figure 4.

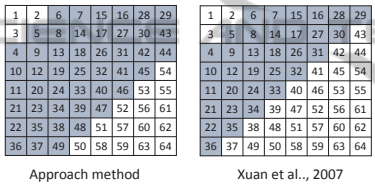


Figure 4: Mosaic range.

2.2.1 Basis Theory (Histogram Pairs)

Xuan et al. proposed a method for reversible watermarking based on the definition of a histogram pair. We turn now to a brief discussion of this approach.

First, we assume that the DCT coefficients take x[a,b], where 'a' and 'b' are the immediately neighboring feature values (b = a+1, a > b, where 'a' and 'b' are integers). Then, the histogram pair is denoted as follows: h=[h_a,h_b], where h_a and h_b are the frequent feature values in an 8x8 DCT coefficients block, given that one of the two frequencies is zero. Therefore, the histogram pair can be applied with the following conditions:

(1) a$\geq$0 and h=[h_a,0] (2) a$<$0 and h=[0,h_a]

Furthermore, when the histogram is not zero, then its original position and the thing that is zero is expanding. Embedding and extracting watermarks proceeds as follows.

(1) a$\geq$0

■ When the bit to embed is '1'. Change one frequent in the original position to the expanding position.

■ When the bit to embed is '0'. Nothing.

(2) a$<$0

■ When the bit to embed is '1'. Change one frequent in the original position to the expanding position.

■ When the bit to embed is '0'. Nothing.

2.2.2 Histogram Expansion

The histogram is expanded to secure a domain for embedding information. In this section, we describe how the histogram is expanded. The expansion proceeds as follows:

■Decide the thresholds T and S.

■If T $\geq$ 0, add 1 to all the values larger than T.

■If T $<$ 0, subtract 1 from all the values smaller than T. The following describes how the expansion of the histogram is inverted.

■ If T $\geq$ 0, subtract 1 from all the values larger than T.

■ If T $<$ 0, add 1 to all the values smaller than T.

2.2.3 Embedding and Extraction

The watermark is embedded and extracted according to the following algorithm.

[Embedding]

(1) Decide the region for embedding in the 8x8 block.

(2) Decide the thresholds T and S.

(3) Expand the histogram.

(4) Embed the information.

(5) Change the threshold T.

■ If T$\geq$0, T changes to -T.

■ If T$<$0, T changes to -T-1.

(6) If the embedding process is incomplete, repeat Steps (3) through (5). Upon reaching the threshold S, the embedding process is complete and the histogram is expanded.

[Extraction]

In order to extract the watermark, the threshold, the embedding region, and the payload must be known. We begin with the threshold S (i.e., the stopping value).

(1) Extract the information.

(2) Inverse the expansion of the histogram.

(3) Change the threshold T.

■ If T$\geq$0, T changes to -T-1.

■ If T$<$0, T changes to -T.

(4) If the extraction is incomplete, Steps (1) through (3) are repeated.

3 PROPOSED METHOD

3.1 Proposed Method

With JPEG compression, our proposed method can remove a mosaic without deteriorating the image. This is accomplished by embedding the information that is needed to restore the original image with reversible watermarking. In addition, unauthorized reconstructions of this information are prevented by encrypting the information needed for removing a mosaic. This ensures that the proposed method protects against the infringement of privacy. In this study, we apply MPEG2-style coding (exclusively to I-frames).

3.1.1 Generating the Mosaic

In order to obtain a JPEG image, the following steps are undertaken.

(1) The original image is divided into 8x8-sized blocks.

(2) The blocks are transformed using DCT.

(3) Blocks are quantized.

(4) VLC encoding is performed.

After quantization, the NxN blocks around the discrete cosine component are set to zero. This distorts the image, providing the mosaic. To remove the mosaic, the original value is set to 0. The method for generating mosaics is illustrated in Figure 5.

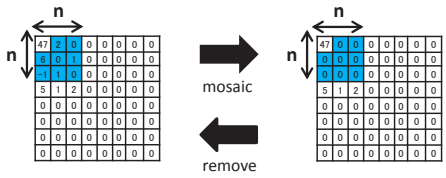


Figure 5: Generation a mosaic (n=3).

3.1.2 Generating Reversible Mosaics

The method for applying a reversible mosaic is shown in Figure 6.

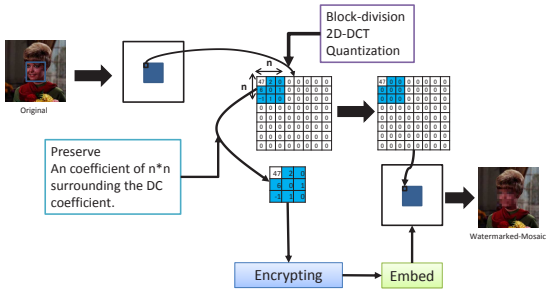


Figure 6: Applying a reversible mosaic.

In order to apply a reversible mosaic, the following steps are required during the JPEG compression. An NxN component is stored around the discrete cosine component after the image is divided into blocks, DCT processing, and quantization. The NxN components are set to zero, with the exception of the discrete cosine component. The information is then embedded and stored as a reversible watermark using the technique proposed by (Xuan et al., 2007). (Assuming a watermarked mosaic picture as follows). Finally, the JPEG-compressed and watermarked mosaic picture is produced with entropy encoding.

3.1.3 Removing a Reversible Mosaic

Figure 7 illustrates the process for removing a reversible mosaic.

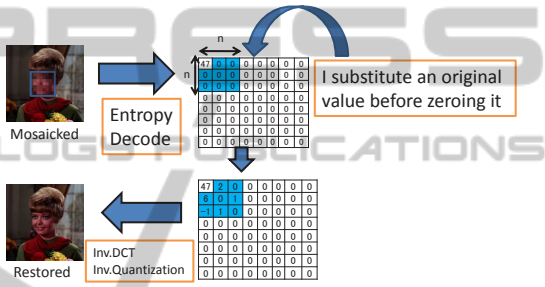


Figure 7: Decrypting and restoring a reversible mosaic.

In order to remove a reversible mosaic, the following steps are required after the JPEG compression. First, entropy decoding is performed on the JPEG-compressed and watermarked mosaic picture. Second, the watermark information is extracted using the technique proposed by (Xuan et al., 2007). Third, the information that was extracted for an NxN component and set to zero is substituted, with the exception of the discrete cosine component. Thus, the mosaic is removed.

3.1.4 Problem

In large-sized images, this method does not conceal areas completely, as shown in Figure 8.

N=1,n=5

Figure 8: Insufficiently concealing a private area.

As shown in Figure 9, we use the same values for multiple blocks in the vicinity of NxN, changing the

particle size of the mosaic. Thus, it is possible to apply a mosaic to a large-sized image.

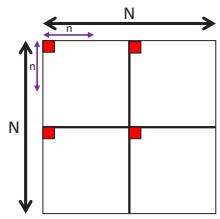


Figure 9: Method for applying a mosaic to a large-sized image.

As shown in Figure 10 , the private are is sufficiently concealed.

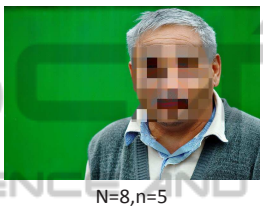


Figure 10: Mosaic successfully applied to a large-sized image.

3.2 Implementation Approach

We combine the two techniques respectively proposed in Sections 2 and 3 for implementation, as shown in Figures 11 and 12.

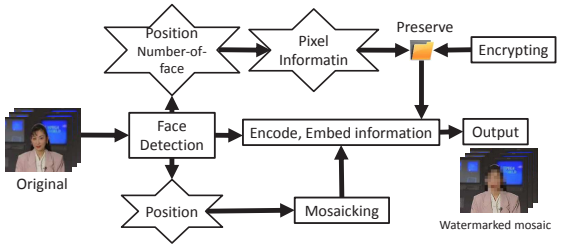


Figure 11: General view of the proposal in terms of applying mosaics and encryption.

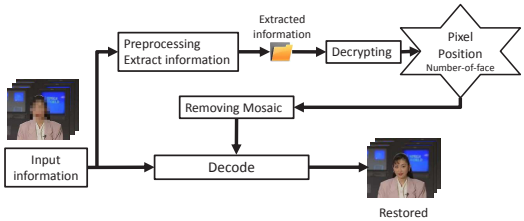


Figure 12: General view of the proposal in terms of decryption and restoration.

To realize a reversible mosaic after MPEG2 compression (with a GOP exclusively for I-frames) four steps are required.

First, the positional information is obtained for the face using the face-detection technique proposed by (Bradski, 1998). Second, the number of faces is determined and the pixel information for these faces is derived based on the positional information. Third, the positional information for all faces is encrypted, along with the pixel information and information regarding the number of faces. Finally, the video is compressed (using MPEG2 compression), embedding the information that was encrypted and generating a reversible mosaic for the facial areas. The following explains the steps for removing the reversible mosaic from the MPEG2-compressed video.

First, the watermark information is extracted and decrypted. Then, the reversible mosaic is removed used three types of information (viz., pixel information, positional information, and the number of faces). In this study, we used 128-bit AES (advanced encryption standard) encryption in CBC (cipher block chaining) mode, applying MPEG2 compression (Hoelzer, 2015).

4 EXPERIMENTAL RESULTS

In this section, we discuss the results from a simulation we conducted to evaluate the proposed method. We used 352288 CIF (common intermediate format) sequences in our study.

Still images from some of the original videos used for the simulation are shown in Figure 13. The experimental results are shown in Figures 14–18. For the experiment, we used 128-bit AES encryption in CBC mode, and we set the thresholds T and S at 190 and 0, respectively. The quality scale was 3.

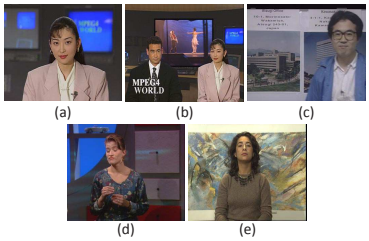


Figure 13: Experimental objects-(a)Akiyo, (b)News, (c)Pamphlet, (d)Sign-Irene, (e)Silent.

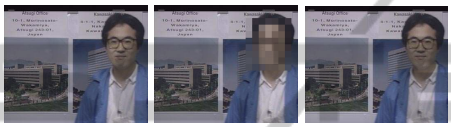
We implemented our proposal in an effort satisfy the three conditions discussed in the Introduction, as seen in Figures 14–18. The mosaicked areas in the image are hidden completely and naturally. Thus, the first condition is met. Furthermore, we calculated the PSNR (peak signal-to-noise ratio) and the SSIM (structural similarity) index for the images from each

(1) (2) (3)

Figure 14: Akiyo - (1)Original, (2)Watermarked mosaic, (3) Restored.

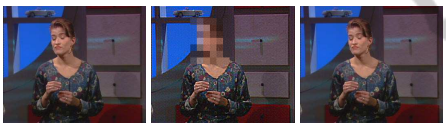
(1) (2) (3)

Figure 15: News - (1)Original, (2)Watermarked mosaic, (3) Restored.



(1) (2) (3)

Figure 16: Pamphlet - (1)Original, (2)Watermarked mosaic, (3) Restored.



(1) (2) (3)

Figure 17: Sign-irene - (1)Original, (2)Watermarked mosaic, (3) Restored.

(1) (2) (3)

Figure 18: Silent - (1)Original, (2)Watermarked mosaic, (3) Restored.

video. The PSNR was infinity, and the SSIM was 1. This result shows that the image is reversible. Thus, these results demonstrate that the proposal meets the second condition. Finally, the information for restoring the mosaicked area was successfully encrypted, satisfying the third condition.

5 CONCLUSIONS

In this paper, we described a technique to facilitate the deterrence of crime with video surveillance while ensuring privacy protection. In future research, we shall consider the introduction of variable-length coding and frames other than I-frames, and we shall aim to increase the processing speed with a hardware implementation.

ACKNOWLEDGEMENTS

The authors would like to thank Junya Yamazaki for his valuable contribution.

REFERENCES

Bradski, G. (1998). Real time face and object tracking as a component of a perceptual user interface. In *Applications of Computer Vision, 1998. WACV '98. Proceedings., Fourth IEEE Workshop on*, pages 214–219.

Carrillo, P., Kalva, H., and Magliveras, S. (2009). Compression independent reversible encryption for privacy in video surveillance. *EURASIP Journal on Information Security*, 2009(1):429581.

Dufaux, F. and Ebrahimi, T. (2008). Scrambling for privacy protection in video surveillance systems. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(8):1168–1174.

Hoelzer, S. Mpeg-2 overview and matlab codec project. http://www.cs.cf.ac.uk/Dave/Multimedia/Lecture_Examples/Compression/mpegproj/ accessed Jan,15,2015.

Li, G., Ito, Y., Yu, X., Nitta, N., and Babaguchi, N. (2009). Recoverable privacy protection for video content distribution. *EURASIP Journal on Information Security*, 2009(1):293031.

Peng, F., wen Zhu, X., and Long, M. (2013). An roi privacy protection scheme for h.264 video based on fmo and chaos. *Information Forensics and Security, IEEE Transactions on*, 8(10):1688–1699.

Saini, M., Atrey, P., Mehrotra, S., and Kankanhalli, M. (2014). W3-privacy: understanding what, when, and where inference channels in multi-camera surveillance video. *Multimedia Tools and Applications*, 68(1):135–158.

Xuan, G., Shi, Y. Q., Ni, Z., Chai, P., Cui, X., and Tong, X. (2007). Reversible data hiding for jpeg images based on histogram pairs. In *Proceedings of the 4th International Conference on Image Analysis and Recognition*, ICIAR'07, pages 715–727, Berlin, Heidelberg. Springer-Verlag.

Yu, X. and Babaguchi, N. (2007). Privacy preserving: Hiding a face in a face. In Yagi, Y., Kang, S., Kweon, I., and Zha, H., editors, *Computer Vision ACCV 2007*, volume 4844 of *Lecture Notes in Computer Science*, pages 651–661. Springer Berlin Heidelberg.