ASePPI, an Adaptive Scrambling enabling Privacy Protection and Intelligibility in H.264/AVC

Natacha Ruchaud
Eurecom
Sophia-Antipolis, France
Email: ruchaud@eurecom.fr

Jean-Luc Dugelay
Eurecom
Sophia-Antipolis, France
Email: dugelay@eurecom.fr

***Abstract*—The usage of video surveillance systems increases more and more every year and protecting people privacy becomes a serious concern. In this paper, we present ASePPI, an Adaptive Scrambling enabling Privacy Protection and Intelligibility. It operates in the DCT domain within the H.264 standard. For each residual block of the luminance channel inside the region of interest, we encrypt the coefficients. Whereas encrypted coefficients appear as noise in the protected image, the DC value is dedicated to restore some of the original information. Thus, the proposed approach automatically adapts the level of protection according to the resolution of the region of interest. Comparing to existing methods, our framework provides better privacy protection with some flexibilities on the appearance of the protected version yielding better visibility of the scene for monitoring. Moreover, the impact on the source coding stream is negligible. Indeed, the results demonstrate a slight decrease in the quality of the reconstructed images and a small percentage of bits overhead.**

I. Introduction

Video surveillance is becoming part of daily life and is a major component of many security systems. While we increasingly use cameras, the resolution of visual sensors (e.g., 4k, HD) and the performance of video processing algorithms (e.g., identity recognition) are continuously increasing. This allows automatic image analysis (e.g. recognition of people, vehicles, animals or bags) in CCTV (Closed-Circuit TeleVision) systems. Detection and recognition systems combined with pervasive networks of dense cameras highlight issues in privacy policy.

Solutions to protect privacy data in surveillance cameras already exist, e.g. using black mask to block out a PIN number entry for ATM security cameras, or to protect private ownership for outdoor security cameras.

However, protecting the privacy of people is more complex given that the monitoring of their actions should not be hampered. Thus, one challenge raised in the article is to manage the trade-off between the privacy protection and the intelligibility (i.e. keeping a fair visualization of the scene).

Naive methods, like blurring, blacking out or pixelization are already used to anonymize people (e.g., Google Street View) but they are not reversible.

Authors in [10], [11], in the spatial domain, shift the Most Significant Bits (MSBs) of encrypted pixels from a RoI (Region of Interest) to the Least Significant Bits (LSBs). Then, the bits from the edge value of the RoI (shape of the body) replace the MSBs of the resulting image in order to keep the scene understandable. This privacy filter is not robust against some manipulations, in particular compression. Nowadays, almost all videos are compressed, therefore, image processing algorithms should be compliant with the compression.

Encryption approaches operate either before (e.g. [1]), during (e.g. [3]) or after (e.g. [14]) compression, denoted pre-, in- and post-compression encryption algorithms. For pre-compression encryption, we will not recover the exact original encrypted values due to the lossy compression therefore we cannot fully decrypt them. Post-compression encryption requires an additional step to make sure that the generated bitstream is decodable by a conventional decoder, but it is too complex and has a little added value. Therefore, our process operates during the compression.

The rest of the paper is organized as follows: in the next section, we summarize the current state-of-the-art of privacy protection techniques using in-compression encryption. In Section 3, we describe the proposed approach. We present and discuss of the results in Section 4. Finally, we draw some conclusions and give an outlook for possible future works in Section 5.

II. Related Works

The most popular current standard for video compression is H.264/AVC. The baseline profile supports Intra (I) and Predicted frames (P) and entropy coding with context-adaptive variable-length codes (CAVLC). I frames contains only intra prediction, intra blocks are predicted from previously coded data within the same frame. P frames contains intra prediction but also inter prediction, inter blocks are predicted from blocks of a previous reference frame. The residual blocks are the differences between the predicted blocks and the real ones.

The following methods including the one of this paper, perform, for each block, a motion compensation in the original video, and then scramble the quantized and DCT-transformed residues.

In H.264/AVC, directly encrypting blocks in the privacy region of a video frame will result in drift error in the non-privacy region due to intra and inter prediction from the privacy region. Therefore, Tong, Dai et al. [3] propose two main methods to prevent such drift error: Mode Restricted Intra Prediction (MRIP) and Search Window Restricted Motion

Estimation (SWRME). To prevent the drift error caused by intra prediction when applying the scrambling on the blocks of the RoI, the fundamental idea in the MRIP technique is to restrict the possible intra prediction modes for blocks around the boundary of the privacy region. The principle of SWRME is to forbid to use any block in the privacy region of the reference frame, to predict a block in the non-privacy region of the current frame.

Coefficient sign scrambling is a common encryption technique widely used within the context of DCT-based compression formats. Dufaux et al. [4] propose to scramble the signs of the nonzero coefficients of each blocks of the privacy region within the MPEG-4 framework. However, it produces a relatively weak scrambling effect especially on high resolution images.

To enhance the scrambling effect for privacy protection, Wang et al. [15] propose to encrypt the intra prediction modes (IPM) in addition to the signs of the nonzero coefficients (SNC) within the privacy region. They also propose a spiral binary mask mechanism to reduce the bitrate overhead incurred by flagging the position of the privacy region. Su and al. [13] directly modify the related data in the H.264/AVC compressed bitstreams while embedding the correct information in the AC coefficients. Khlif and al. [6] scramble the signs of motion vectors using chaotic cryptography algorithm.

Contrary to [4], these three previous methods produce a strong scrambling effect yielding to noisy pictures which hamper the monitoring. Encryption or scrambling are powerful reversible methods to protect the privacy but they have issues to manage the trade-off with the intelligibility.

Ruchaud et al. [12] handle this trade-off by applying a bitwise XOR operation between each DCT (DC+AC) coefficient and pseudo-random numbers within the JPEG framework (operating on still images, not on videos). Thus, they shift down the encrypted coefficients from one position which allows the insertion of a value of their choice into the DC of each block enabling a better visualization of the decompressed privacy-protected images.

We take over the idea of setting the DC to a value of our choice to control the final appearance of the images while encrypting the original coefficients to protect the privacy. We make this compliant with H.264/AVC. In addition, our process automatically adapts the strength of the protection according to the size of the privacy-sensitive region which is not the case for the existing methods.

III. ASePPI, an Adaptive Scrambling enabling Privacy Protection and Intelligibility

We integrate our approach only for the luminance channel (Y) within the residual blocks produced by the H.264/AVC framework. Indeed, operating on the chroma channels yields to unpleasant colors. To avoid drift error produced by our process, we applied MRIP and SWRME approaches. The code of our proposed process is available on a Github website [1].

[1]https://github.com/NatachaRuchaud/ASePPI

A. The region of interest (RoI)

Every 10 frames, we previously annotated or automatically detect the region of interests (e.g., people faces and bodies), denoted RoIs. We use publicly available face and people detectors. The position of the RoI contains the upper left point and the size (four numbers). We compute a bitwise XOR operation among each number and a random number (RN) generated by a pseudo-random sequence controlled by a secret key. The encrypted RoI position is not encoded into the scrambled video stream, we store it independently from the privacy protected video. The number of bits needed to store the RoI position is negligible. For instance, for a 4K resolution (i.e., 4096*2160 pixels), we use 12 bits to store each encrypted number, thus 48 bits every 10 frames (i.e., 4.8 each frame).

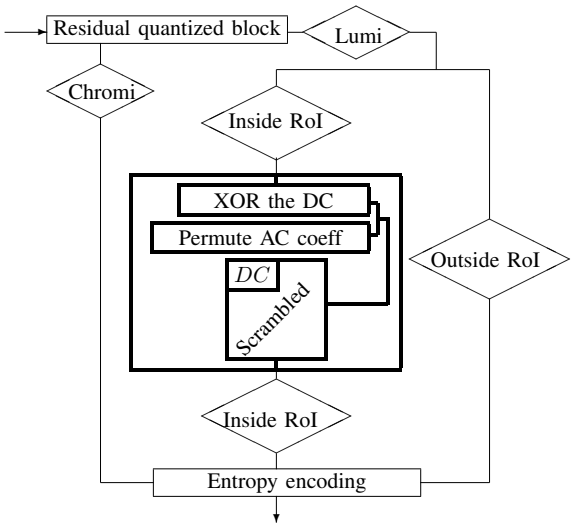

Fig. 1: Workflow of the process for one intra block of I frame.

B. Encrypting the residual of I frames blocks

Each intra prediction inside the RoI follows additional steps illustrated in bold in Figure 1. We encrypt the DC and AC coefficients to protect data information. Then, the encrypted coefficients are shifted to make available the DC position. Thus, we lost the least significant coefficient (the last AC).

1) Encrypting the DC: in order to limit the noise due to the scrambled DC which is hidden in the first AC coefficient, we encrypt as in the algorithm 1, with sign(DC) equal to -1 if the DC sign is negative and +1 otherwise.

2) Permute the AC: we extract the AC coefficients except the last one according to the zigzag code. Note, p, the number of AC coefficients before EOB (End-of-Block, the remaining coefficients are zero). To scramble them, we randomly permute the $p-1$ AC coefficients using the Knuth shuffle algorithm [2] that re-arranges their order. In other words, the AC coefficients before the last non-zero coefficient are randomly permuted. The last non-zero coefficient is used to mark the end of the permutation.

3) Hidding scrambled coefficients into the AC ones: we shift the scrambled coefficients by one position towards the high frequencies in order to make available the DC position.

if *($|DC| < 16$)* **then**
  X = 16;
else
  $X = 2^n$;

Generate a random number (RN) as in III-A;
if *($DC \neq 0$) & (($|DC| \neq (RN \mod X)$))* **then**
  DC_e = ($|DC| \oplus (RN \mod X)$)*sign(DC);
else
  DC_e = DC;

with $n = \lfloor \log_2 |DC| \rfloor$ an integer

Algorithm 1: DC encryption

Then, we re-insert the scrambled coefficients into a block according to the zigzag code and choose the DC value, denoted DC_{new}, with the formula defined in III-B4.

4) Choice of the DC_{new} value: whereas encrypted coefficients appear as noise in the protected image, the DC_{new} value is dedicated to restitute a minimum of information (e.g. the average luminance associated with one or a group of blocks).

Keeping the original DC only for each n*m block leads to a pixelated image of size n*m. The goal is to minimize the size of these blocks in order to preserve the intelligibility while minimizing the performances of face recognition for any resolution. Minimize the size of the blocks is equivalent to maximize the number of blocks. The equation (1) represents the relation between the size of the blocks, denoted S, and the number of blocks, denoted Nb, depending on the number of pixels (h x w) inside the RoI. For example, if S is equal to 24, the residual blocks inside the 24*24 block have the same DC coefficient, which is the DC of the 24*24 block (i.e. the mean of the 24*24 block).

$$Nb = \frac{h * w}{S * S} \tag{1}$$

The higher Nb (i.e. the higher is the image quality), the better the recognition is in general. Our goal is to find the maximum value of Nb to preserve as much as possible the intelligibility while minimizing the performance of face recognition. Therefore, to fulfil this purpose, we did the following empirical study by fixing several values.

We have selected as a baseline the face recognition algorithm Eigen described in [7] based on the Euclidean distance because of its robustness to pixelated face images (compared to some other descriptors). We randomly selected a subset of the Feret [9] and the ScfaceData [5] databases for the training and another one for the testing. We tested this face recognition algorithm on the original images from the testing set and on the pixelated versions of them with different parameter values (i.e. S, h and w).

According to Table I, we have selected the S values associated with the highlighted boxes representing the most important drop in recognition performance at each resolution. From these results and the equation (1), we deduce the maximum Nb which is 99 (e.g. $\frac{176*144}{16*16} = 99$). Face recognition performance significantly drop if Nb is equal to 99 or less. Note that the databases contain almost no variation (e.g. similar lightening conditions, without delay between sessions) compared to the reality, that is why the recognition task still performs upper than 20 % even after a strong pixelization. To have a more accurate value of $max(Nb)$, we should use a more realistic database.

S \ *h*w*	128 x 96	176 x 144	352 x 288	704 x 576
Original	95.6	96	96.4	96.4
8	68.4	76.41	85.4	86
12	22.9	66.5	76.4	85.5
16	20.7	21.3	75.9	84.4
28	5.5	18.8	58.02	77.9
32	4.2	12.7	20.8	75.4
36	3.6	8.5	20.4	73.7
60	0	0	9.6	50.47
64	0	0	8.1	20.2
68	0	0	5.5	19.5

TABLE I: Accuracy of identity recognition (%) from faces.

Looking for maximizing Nb, the relation can be rewritten as in equation 2. S is rounded to its nearest multiple of 4 as in equation (3) because the size of each residual block is 4*4. Therefore, the equation (3) automatically defines S, a multiple of 4, maximizing the number of blocks such as we protect the privacy. However, we can change the value of $max(Nb)$ to have stronger or weaker protection.

$$\begin{aligned} max(Nb) &\geq Nb \\ \Leftrightarrow max(Nb) &\geq \frac{h * w}{S * S} \\ \Leftrightarrow S &\geq \sqrt{\frac{h * w}{max(Nb)}} \end{aligned} \tag{2}$$

$$\begin{aligned} S \approx \left\lceil \frac{\sqrt{\frac{h*w}{max(Nb)}}}{4} \right\rceil * 4 \geq \sqrt{\frac{h * w}{max(Nb)}} \\ S = \left\lceil \frac{\sqrt{\frac{h*w}{99}}}{4} \right\rceil * 4 \end{aligned} \tag{3}$$

C. Encrypting the residual of P frames blocks

In addition to process the transformed and quantized residues of I frames, we also encrypt the ones of P frames. Indeed, in H.264/AVC framework, the blocks inside the RoI may be predicted from unscrambled blocks or become closer to the original one if the reference scrambled blocks are already close to the original ones.

Therefore, we randomly permute the AC coefficients as in III-B2 and leave the DC as it is. Thus, no information is lost.

IV. Experimental Results

We compare the proposed method (ASePPI), with the encryption of the signs of the non-zero coefficients (SNC) and with the addition of the encryption of the intra prediction modes (SNC+IPM) within the privacy region. We apply all methods only on the luminance channel to be comparable to our approach. We use different values of QP and IP in our evaluations. QP is the quantization parameter and IP the intra period that defines the frames number between two I frames.

For the evaluation, we have selected the following sequences: 'hall', 'foreman', 'suzie', 'akiyo', 'carphone', 'claire' and 'miss-america'. In IV-A, we assess the privacy protection and the intelligibility for the scrambled frames of the sequences with CIF size. In IV-B, we also evaluate the bits overhead and the reduction in PSNR performances for the reconstructed sequences with QCIF size.

A. Privacy protection vs Intelligibility

From a subjective point of view, the details of the face is still easily identified applying SNC (see 2(b) and 2(f)) compared to the two others methods (SNC+IPM and ASePPI).

We apply the Eigen face recognition algorithm (same than in III-B4) adding, in the training set, faces from the odd frames for each sequence in CIF size. We test faces from the even frames for each sequence and we get 99.6 % of accuracy for original face images. We report the accuracy of SNC and ASePPI methods in the Table II. Therefore, the results show that our method enhances the privacy protection (of 14.22 % in average) especially when IP increases, compared to SNC.

TABLE II: Accuracy of face recognition depending on the Intra Period (IP), with QP = 24 and QP = 18, respectively.

IP	SNC	ASePPI
5	19.18	9.3
10	20.2	8.6
30	21.1	7.9
50	28.3	7.2

IP	SNC	ASePPI
5	19.9	10.7
10	21.2	9.2
30	22	8.5
50	31	7.7

Nevertheless, using both SNC and IPM hampers the global understanding of the scene. For example, in Figure 2(k) it is not obvious that the protected area contains a person carrying her purse whereas in Figure 2(l) the shape of the head and feet are clearly distinguishable as well as the bag. We also evaluate the intelligibility with two metrics, the peak signal-to-noise ratio (PSNR) to measure the amount of the degradation and the edge similarity score (ESS) [8] to assess the degree of resemblance of the edge and contour information between two images. We apply these metrics between the original RoI and the scrambled RoI of the seven sequences for each QP =18, 24 and IP=1, 5, 10, 30. Compared to ASePPI, SNC+IPM degrades, in average, 8.5 % more and its degree of resemblance of the edge is 22.8 % less important. However, in extreme cases, a block can have only one AC coefficient with a larger amplitude than the DC_{new}. This produces more noise than usual. To avoid these cases, we suggest to use QP lower or equal to 24.

Thus, our proposed approach, ASePPI, produces enough scrambling to protect the privacy of people for different resolutions while it still preserves a fair visualization of the scene which is very important in video surveillance.

B. Impact on source coding stream

The bits overhead is the percentage of bits added by our process comparing to the baseline profile (H.264 without encryption). For example, for the 'foreman' sequence, with QP = 24 and IP = 10, the number of saving bits are 83289 for the baseline profile and 90522 with the integration of our process which produces $100 - 100 * 83289/90522$ % of bits overhead, i.e. 7.99 %. The I frames produce the most important increase of the number of bits in the stream. That is why, the higher the IP the lower is the bits overhead.

TABLE III: Bits overhead (%) with QP set to 24

IP	Suzie	Foreman	Hall	Akiyo	Carphone	Claire	Miss America
1	13.21	8.4	5.39	5.75	2.79	9.96	4.33
5	13	8.2	5.31	5.3	1.91	9.55	3.27
10	12.96	7.99	5.26	4.21	1.72	9.18	3.18
30	12.14	7.43	5.1	3.74	1.63	8.1	2.9

The drop of PSNR performances (for RGB channels) in percentage for reconstructed images (recovered by applying the reverse process with the correct secret key) compared to the original ones, is computed in the same way than the bits overhead. The higher QP the lower is the number of blocks where the last coefficient is lost, thus, the closer are the PSNR performances of our process with the ones of the baseline.

TABLE IV: PSNR decrease (%) with IP set to 10.

QP	Suzie	Foreman	Hall	Akiyo	Carphone	Claire	Miss America
12	1.36	1.32	1.36	1.4	1.8	1.9	0.5
18	0.91	1.01	1.11	1.3	1.3	1.8	0.45
24	0.86	1.26	1.28	1.18	1.76	1.11	0.41
30	0	0	0.7	0	0.14	0.51	0.09

C. Replacement attack

In case of replacement attack, for SNC and ASePPI methods, only the DC of each block is available because all encrypted coefficients are set to 0. This leads to have pixelated images. In SNC case, the size of this pixelization is always the same and small (e.g. 4*4). Thus, for RoI of high resolution, SNC method may fail to protect the privacy whereas ASePPI automatically adapts this size depending on RoI resolution.

V. Conclusion

Contrary to existing methods, the application of ASePPI enhances the scrambling effect for privacy region protection of videos with the aim of keeping the minimum of information required by the surveillance. Our approach automatically adapts the strength of the privacy protection depending on the resolution of the privacy-sensitive regions. The quality of the reconstructed videos are very closed to the original ones and the process produces a small percentage of bits overhead.

Fig. 2: With CIF size, QP= 24 and IP = 5: (a) The 1st frame of original 'foreman' (I frame), (b) encrypted by SNC, (c) encrypted by SNC+IPM, (d) encrypted by our process. (e) The 15th frame of original 'foreman' (P frame), (f) encrypted by SNC, (g) encrypted by SNC+IPM, (h) encrypted by our process. (i) The 40th frame of original 'hall' (P frame), (j) encrypyted by SNC, (k) encrypyted by SNC+IPM, (l) encrypyted by our process.

We are evaluating the security of our process in terms of brute force and replacement attacks and we are refining the method to be robust against these attacks.

As further works, we can subjectively evaluate the efficiency of the privacy protection and the intelligibility by doing a survey, asking the gender, age or ethnicity and the activities of persons where their images are protected by our method.

REFERENCES

[1] T. E. Boult. Pico: Privacy through invertible cryptographic obscuration. In *Computer Vision for Interactive and Intelligent Environment, 2005*, pages 27–38. IEEE, 2005.

[2] D. Chafaï and F. Malrieu. Permutations, partitions, et graphes. In *Recueil de Modèles Aléatoires*, pages 57–68. Springer, 2016.

[3] F. Dai, L. Tong, Y. Zhang, and J. Li. Restricted h. 264/avc video coding for privacy protected video scrambling. *Journal of Visual Communication and Image Representation*, 22(6):479–490, 2011.

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

[5] M. Grgic, K. Delac, and S. Grgic. Scface–surveillance cameras face database. *Multimedia tools and applications*, 51(3):863–879, 2011.

[6] N. Khlif, T. Damak, F. Kammoun, and N. Masmoudi. Motion vectors signs encryption for h. 264/avc. In *Advanced Technologies for Signal and Image Processing (ATSIP), 2014 1st International Conference on*, pages 1–6. IEEE, 2014.

[7] V. Kshirsagar, M. Baviskar, and M. Gaikwad. Face recognition using eigenfaces. In *Computer Research and Development (ICCRD), 2011 3rd International Conference on*, volume 2, pages 302–306. IEEE, 2011.

[8] Y. Mao and M. Wu. A joint signal processing and cryptographic approach to multimedia encryption. *IEEE Transactions on Image Processing*, 15(7):2061–2075, 2006.

[9] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE Transactions on pattern analysis and machine intelligence*, 22(10):1090–1104, 2000.

[10] N. Ruchaud. Privacy protection filter using stegoscrambling in video surveillance. In *MediaEval*, 2015.

[11] N. Ruchaud and J. L. Dugelay. Efficient privacy protection in video surveillance by stegoscrambling. In *WIFS 7th IEEE International Workshop on Information Forensics and Security*, 2015.

[12] N. Ruchaud and J.-L. Dugelay. Privacy protecting, intelligibility preserving video surveillance. In *Multimedia & Expo Workshops (ICMEW), 2016 IEEE International Conference on*, pages 1–6. IEEE, 2016.

[13] P.-C. Su, W.-Y. Chen, S.-Y. Shiau, C.-Y. Wu, and A. Y. Su. A privacy protection scheme in h. 264/avc by data hiding. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*, pages 1–7. IEEE, 2013.

[14] A. Unterweger, K. Van Ryckegem, D. Engel, and A. Uhl. Building a post-compression region-of-interest encryption framework for existing video surveillance systems. *Multimedia Systems*, 22(5):617–639, 2016.

[15] Y. Wang, F. Kurugollu, et al. Privacy region protection for h. 264/avc with enhanced scrambling effect and a low bitrate overhead. *Signal Processing: Image Communication*, 35:71–84, 2015.