SCALABLE CODING OF STREAM CIPHER ENCRYPTED IMAGES VIA ADAPTIVE SAMPLING

Jiantao Zhou[1] and Oscar C. Au[2]

[1]Department of Computer and Information Science, University of Macau
[2] Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology

ABSTRACT

This work proposes a novel scalable image compression method for stream cipher encrypted images. The bit stream in the base layer is produced by coding a series of non-overlapping patches of the uniformly down-sampled version of the encrypted image. An off-line learning approach can be exploited to model the reconstruction error of original image patch based on the intrinsic relationship between the local complexity and the length of the compressed bit stream. This error model leads to a greedy strategy of adaptively selecting pixels to be coded in the enhancement layer. At the decoder side, an iterative, multi-scale technique is developed to reconstruct the image from available pixel samples. Experimental results demonstrate that the proposed scheme outperforms the state-of-the-art in terms of rate-distortion (RD) performance at low and medium rate regions.

Index Terms— Image compression over encrypted domain, scalable image coding

1. INTRODUCTION

The standard way of providing data security is to encrypt it via cryptographic algorithms, e.g., AES and RC4. In many practical scenarios such as cloud computing, the parties who process the encrypted data are often untrusted, and hence have no access to the secret key. This has led to the challenging problem of how to achieve superior efficiency of processing the encrypted data with zero knowledge of the secret.

As one of the most common operations on multimedia data, multimedia compression over encrypted domain has received increasing attention since the last decade. Johnson *et. al* showed that the stream cipher encrypted data is compressible through the use of coding with side information principles, without compromising either the compression efficiency or the information-theoretic security [1]. By applying LDPC codes in various bit-planes and exploiting the inter/intra correlation, Lazzeretti and Barni presented several methods for lossless compression of encrypted grayscale/color images [2]. Furthermore, Kumar and Makur extended the approach of [1] to the prediction error domain and achieved better lossless compression performance on the encrypted grayscale/color images [3]. Aided by rate-compatible punctured turbo codes, Liu *et. al* developed a progressive method to losslessly compress stream cipher encrypted grayscale/color images [4].

To achieve higher compression ratios, lossy compression of encrypted images was also studied [5–8]. Zhang *et. al* proposed a scalable lossy coding framework of encrypted images via a multi-resolution construction [5], where encryption is achieved by *adding* (modulo-256) the pixel values with the key stream. For images encrypted by standard stream cipher (bitwise XOR), Kang *et. al* designed a new scalable coding scheme through transmitting a down-sampled version of the encrypted image as the base layer, then selectively choosing additional bit planes to be sent in the enhancement layer [6]. Further, Zhang designed an image encryption scheme via pixel-domain permutation, and demonstrated that the encrypted file can be efficiently compressed by discarding the excessively rough and fine information in the transform domain [7]. We recently also suggested a high performance coding approach for encrypted images, achieving nearly the same compression efficiency as a conventional image coding system having original, unencrypted images as input [8].

In this work, we focus on encrypted images obtained by applying stream cipher in the standard format (e.g., AES-CTR [9]); namely, ciphertext is generated by XORing the plaintext with the key stream. We propose a novel scalable coding scheme for encrypted images. The bit stream in the base layer is produced by coding a series of non-overlapping patches of the uniformly down-sampled version of the encrypted image. An off-line learning approach can be exploited to model the reconstruction error of image patch based on the intrinsic relationship between the local complexity and the length of the compressed bit stream. This error model leads to a greedy strategy of adaptively selecting pixels to be coded in the enhancement layer. At the decoder side, an iterative, multi-scale technique is developed to reconstruct the image from available sample pixels. Experimental results demonstrate that the proposed scheme outperforms the state-of-the-arts in terms of RD performance at low and medium rate regions.

This work was supported by the Macau Science and Technology Development Fund under grant FDCT/009/2013/A1 and the Research Committee at University of Macau under grant SRG023-FST13-ZJT, MYRG2014-00031-FST, and MRG021/ZJT/2013/FST.

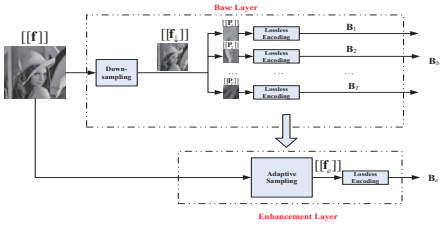

Fig. 1. The proposed scalable image coding scheme.

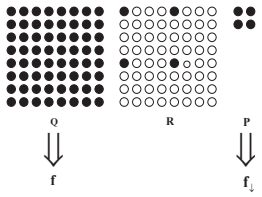

Fig. 2. The relationship among $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$, where $M = 2$.

2. SCALABLE CODING OF ENCRYPTED IMAGES VIA ADAPTIVE SAMPLING

When stream cipher is used in the standard format, the encrypted image is obtained by

$$[[\mathbf{f}]] = \mathbf{f} \oplus \mathbf{K} \tag{1}$$

where $\mathbf{f}$ and $[[\mathbf{f}]]$ denote the original and the encrypted images of size $N \times N$, respectively, and $\mathbf{K}$ is the secret key stream. Here, $\oplus$ represents the bitwise XOR operator, and all images are assumed to be 8-bits. Throughout the paper, we use $[[\mathbf{x}]]$ to represent the encrypted version of $\mathbf{x}$.

To efficiently compress the encrypted image $[[\mathbf{f}]]$, we develop a novel scalable image coding framework operated over the encrypted domain. In Fig. 1, we show the schematic diagram of the proposed scheme, with a base layer and an enhancement layer, where the original image is given just for better illustration purpose.

2.1. Base layer encoding

To generate the bit stream in the base layer, we first down-sample $[[\mathbf{f}]]$ into $[[\mathbf{f}_\downarrow]]$. Here, we stick to conventional square pixel grid by uniform spatial down sampling of $[[\mathbf{f}]]$. Out of practical considerations, we make a more compact representation of $[[\mathbf{f}]]$ by decimating every four rows and every four columns, namely, the resulting $[[\mathbf{f}_\downarrow]]$ is of size $N/4 \times N/4$. This simple down-sampling strategy can be readily implemented in the encrypted domain, as the spatial relationship among pixels keeps intact after encryption. Further, such down-sampling offers an important operational advantage: $[[\mathbf{f}_\downarrow]]$ still remains a uniform rectilinear grid of pixels, making it readily compressible by *any* existing techniques for compressing encrypted images, e.g. [4].

Instead of coding $[[\mathbf{f}_\downarrow]]$ as a whole, we propose to partition it into a series of non-overlapping patches $[[\mathbf{P}_t]]$ of size $M \times M$. Each $[[\mathbf{P}_t]]$ is then treated as a subimage, and coded individually into a binary bit stream $\mathbf{B}_t$, by applying the lossless compression method of [4]. The final bit stream in the base layer $\mathbf{B}_b$ is formed by concatenating all $\mathbf{B}_t$'s, i.e.

$$\mathbf{B}_b = \mathbf{B}_1 \mathbf{B}_2 \cdots \mathbf{B}_T \tag{2}$$

where $T = \frac{N^2}{16M^2}$.

A free byproduct offered by the base layer encoding is the length of each $\mathbf{B}_t$, denoted by h_t. As also mentioned in [8], h_t, called local complexity indicator (**LCI**), provides critical information concerning the local complexity of $\mathbf{P}_t$. Smooth patches that are more compressible would produce small values of h_t, while those high-activity regions typically generate large h_t. Such crucial information accessible in the encrypted domain without extra cost will be shown to be useful in guiding the encoding of the enhancement layer through a context-adaptive mechanism.

2.2. Context modeling of patch reconstruction error

Let $\mathbf{P}$ be a generic $M \times M$ sized patch in $\mathbf{f}_\downarrow$, and $\mathbf{Q}$ be the associated $4M \times 4M$ sized patch in $\mathbf{f}$. In other words, $\mathbf{P}$ is obtained by down-sampling $\mathbf{Q}$ by a factor of 4. Denote $\mathbf{R}$ as the $4M \times 4M$ version of $\mathbf{P}$ by duplicating the pixels of $\mathbf{P}$ in their original grids, while leaving the remaining locations empty. The relationship among $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$ can be illustrated in Fig. 2, where solid dots denote the available pixels, while the hollow ones denote the vacant locations.

Let $\hat{\mathbf{Q}}(n)$ be the reconstructed image patch from the pixel samples of $\mathbf{R}$ and additional $n \times s$ samples randomly selected from $\mathbf{Q}$ corresponding to the empty locations of $\mathbf{R}$, where $s = 128$ is a constant step size. The discussion on the image reconstruction approach from pixel samples is deferred to Section 3. In the context of our scalable image coding scheme, the samples in $\mathbf{R}$ are provided by the base layer, while the additional samples are made available by the enhancement layer.

From the perspective of sampling theory, additional samples allocated in the smooth regions help less in reducing the reconstruction errors, compared with those in complex regions. This is because, in smooth regions, the existing samples available through base layer might be sufficient to guarantee faithful reconstructions, making the additional samples less useful. Therefore, when generating the bit stream in the enhancement layer, it is more advantageous to code more samples in those complex regions, so as to achieve better overall RD performance.

To develop an appropriate strategy for selecting pixels to be coded in the enhancement layer, we need to have a deeper understanding about the relationship between the reconstruction error and the local complexity. In the following, we

establish a context model for the patch reconstruction error based on the observable **LCI** h. This error model can be shown to be able to derive a greedy strategy for pixel sample selection in the enhancement layer. To this end, we first define the reconstruction error difference (**RED**) by

$$\begin{aligned} d(n) &= e(n-1) - e(n) \\ &= ||\hat{\mathbf{Q}}(n-1)) - \mathbf{Q}||_2^2 - ||\hat{\mathbf{Q}}(n) - \mathbf{Q}||_2^2 \end{aligned} \quad (3)$$

where $e(n)$ denotes the patch reconstruction error when $n \times s$ additional pixel samples are given. The value of $d(n)$, which is non-negative, reflects the effectiveness of increasing the number of additional samples from $(n-1) \times s$ to $n \times s$.

As explained above and also verified experimentally, $d(n)$ strongly correlates to the local smoothness of the patch. For those with smooth signal waveforms, $d(n)$ tends to be small, while for those high-activity regions, it becomes large. We hence introduce the conditional probability $p(d(n)|h)$ as a statistical context model of $d(n)$. To prevent the problem of context dilution caused by too many modeling parameters, we propose to estimate the conditional expectation $\mathbb{E}(d(n)|h)$, rather than $p(d(n)|h)$ itself. Still, the space spanned by h is too large in practice. To further narrow the context space, we apply uniform scalar quantization on h with step size τ, namely, $\tilde{h} = \lfloor h/\tau \rfloor$.

To estimate $\mathbb{E}(d(n)|\tilde{h})$, we employ an off-line learning approach with the assistance of $U = 100$ training images. For each training image $\mathbf{f}_u$, where $1 \le u \le U$, the patches $\mathbf{Q}_{u,t}$, $\mathbf{R}_{u,t}$, $\mathbf{P}_{u,t}$ can be readily extracted. For each $\mathbf{R}_{u,t}$, we increment $n \times s$ pixel samples in the vacant locations, and get the reconstructed patch $\hat{\mathbf{Q}}_{u,t}(n)$ by the technique to be presented in Section 3. The corresponding $d_{u,t}(n)$ can then be calculated by (3). Eventually, each patch $\mathbf{R}_{u,t}$ is associated with a **RED** vector $\mathbf{d}_{u,t} = (d_{u,t}(1), d_{u,t}(2), \cdots, d_{u,t}(S))$, where

$$S = \frac{(4M)^2 - M^2}{s} = \frac{15M^2}{s} \quad (4)$$

Meanwhile, the value of $\tilde{h}_{u,t}$ can be obtained by compressing $[[\mathbf{P}_{u,t}]]$ with the method of [4] and measuring the length of the compressed bit stream. Then, the conditional expectation $\mathbb{E}(d(n)|\tilde{h})$ can be computed by

$$\mathbb{E}(d(n)|\tilde{h} = h_o) = \frac{\sum_{(u,t)\in\Omega} d_{u,t}(n)}{|\Omega|} \quad (5)$$

where $\Omega = \{(u,t)|\tilde{h}_{u,t} = h_o\}$.

It should be noted that the above training process is conducted off-line, and hence the encoding complexity will not be materially increased, as all the pre-calculated conditional expectations can be implemented in a look-up table.

(a) $F = 10$ (b) $F = 50$ (c) $F = 100$ (d) $F = 200$

Fig. 3. Sample selection for different values of F.

2.3. Enhancement layer coding via greedy strategy

With the above context model of patch reconstruction error, we in this subsection propose a greedy algorithm to strategically select the pixels to be coded in the enhancement layer. Let $\tilde{\mathbf{h}} = (\tilde{h}_1, \tilde{h}_2, \cdots, \tilde{h}_T)'$ be the vector recording the LCI's of all the $4M \times 4M$ sized patch of the encrypted image. It is assumed that totally there are $F \times s$ samples to be coded in the enhancement layer, where F is a parameter controlling the coding rate. The greedy algorithm for the pixel sample selection is given as follows:

Step 1: Initialize a vector $\mathbf{E} \triangleq (E_1, E_2, \cdots, E_T)' = (\mathbb{E}(d(n_1)|\tilde{h}_1)), \mathbb{E}(d(n_2)|\tilde{h}_2)), \cdots, \mathbb{E}(d(n_T)|\tilde{h}_T))'$ by setting all n_j's to be 1, where the conditional expectations are obtained from the off-line training process.

Step 2: Find the index q corresponding to the maximum element of $\mathbf{E}$, i.e.,

$$q = \underset{1\le i\le T}{\operatorname{argmax}} E_i \quad (6)$$

When multiple q's exist, we simply choose the minimum one.

Step 3: Randomly choose s encrypted pixels from $[[\mathbf{Q}]]_q$ that have never been selected, according to a *public* seed.

Step 4: Update

$$E_q = \mathbb{E}(d(n_q + 1)|\tilde{h}_q)), n_q = n_q + 1 \quad (7)$$

Step 5: Repeat Steps 2-5 for F times until all $F \times s$ samples are selected.

Step 6: Reshape the selected pixel samples into a 2-D image $[[\mathbf{f}_e]]$, and encode it into a binary bit stream $\mathbf{B}_e$ using the method of [4].

Fig. 3 demonstrates the results of sample selection when $F = 10, 50, 100, 200$, where the original image is given just for better illustration purpose. It can be observed that most of the samples are located at the complex foreground, while the sky in the background with homogenous characteristics has slim chance of getting samples even when $F = 200$. This validates the effectiveness of our proposed sample selection algorithm operated over encrypted domain.

3. IMAGE RECONSTRUCTION

Upon receiving the bit stream $\mathbf{B}_b$ from the base layer, the decoding algorithm of [4] can be applied to get $[[\mathbf{f}_\downarrow]]$. As the

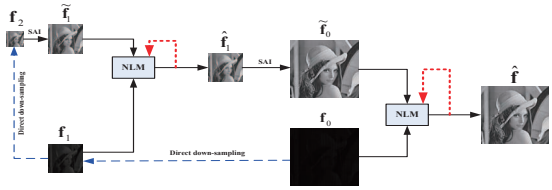

Fig. 4. Schematic diagram of image reconstruction.

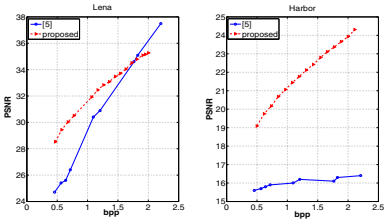

Fig. 5. Comparison of RD performance.

uniform down-sampling rule is public, $[[\mathbf{f}_{\downarrow}]]$ can be appropriately decrypted into $\mathbf{f}_{\downarrow}$ by XORing with the corresponding key stream. In the case that the enhancement layer is not available, $\mathbf{f}_{\downarrow}$ can be directly up-converted to $\hat{\mathbf{f}}_b$, using the method to be presented below. In some application scenarios, $\hat{\mathbf{f}}_b$ can be utilized as the preview version for customers.

When the bit stream of the enhancement layer arrives, the decoding algorithm of [4] can be similarly employed to obtain $[[\mathbf{f}_e]]$, where the base layer reconstruction $\hat{\mathbf{f}}_b$ serves as the side information. As the encoder and the decoder can be perfectly synchronized, the key stream can be appropriately extracted to decrypt $[[\mathbf{f}_e]]$ into $\mathbf{f}_e$.

With $\mathbf{f}_{\downarrow}$ and $\mathbf{f}_e$, the decoder aims to re-estimate the original image. We propose an iterative, multi-scale technique to reconstruct the original image, as depicted in Fig. 4. Let $\mathbf{f}_0$ be the image containing all the available samples from $\mathbf{f}_{\downarrow}$ and $\mathbf{f}_e$, while the vacant locations are padded with 0's. In our proposed multi-scale framework, $\mathbf{f}_0$ is successively down-sampled twice by a factor 2 respectively to form a pyramid: $\mathbf{f}_1$ and $\mathbf{f}_2$, by replacing each low-resolution pixel with the average of existing high-resolution ones. The image reconstruction starts from the lowest level 2. We first up-convert $\mathbf{f}_2$ to a higher level $\tilde{\mathbf{f}}_1$ via a parametric-model based interpolation method. Due to the excellent interpolation performance and modest computational complexity, the soft-decision adaptive interpolation (SAI) [10] is adopted. With $\tilde{\mathbf{f}}_1$ and $\mathbf{f}_1$, we then attempt to get a refined version by estimating the missing pixels of $\mathbf{f}_1$. To this end, we employ a non-local mean (NLM)-based approach, in which all the weighting vectors are computed from the up-converted $\tilde{\mathbf{f}}_1$. In other words, $\tilde{\mathbf{f}}_1$ now serves as the prior knowledge for the estimation task. Specifically, the ith missing pixels in $\mathbf{f}_1$ are estimated by

$$\hat{\mathbf{f}}_1(i) = \sum_{j \in \mathcal{W}_i} w(i,j)\mathbf{f}_1(j) \tag{8}$$

where $\mathcal{W}_i$ denotes a local window centered at pixel i. The family of weights $w(i,j)$'s depend on the similarity between the pixels i and j, and are explicitly given by

$$w(i,j) = \frac{1}{Z(i)} e^{\frac{-||\tilde{\mathbf{f}}_1(\mathcal{N}_i) - \tilde{\mathbf{f}}_1(\mathcal{N}_j)||^2_{2,\alpha}}{\delta^2}} \tag{9}$$

where the gray level vectors $\tilde{\mathbf{f}}_1(\mathcal{N}_i)$ and $\tilde{\mathbf{f}}_1(\mathcal{N}_j)$ represent two square neighborhoods of fixed size centered at pixels i and j, respectively. Here, $||\cdot||^2_{2,\alpha}$ denotes the weighted Euclidean distance, with $\alpha > 0$ being the standard deviation of the Gaussian kernel, and $Z(i)$ is a normalizing constant

$$Z(i) = \sum_j e^{\frac{-||\tilde{\mathbf{f}}_1(\mathcal{N}_i) - \tilde{\mathbf{f}}_1(\mathcal{N}_j)||^2_{2,\alpha}}{\delta^2}} \tag{10}$$

where δ controls the degree of filtering.

To further enhance the performance, we iterate the above procedure by replacing $\tilde{\mathbf{f}}_1$ with the newly refined $\hat{\mathbf{f}}_1$. Heuristically, we find that 2-3 iterations are sufficient to produce satisfactory results. The resulting $\hat{\mathbf{f}}_1$ is then up-converted to $\tilde{\mathbf{f}}_0$ via SAI. At level 0, similar iterative NLM-based refinement process can be carried out to get the final estimate $\hat{\mathbf{f}}$.

4. EXPERIMENTAL RESULTS

To demonstrate the performance of our proposed scalable coding scheme for stream cipher encrypted images, we compare it with the method of [5] in terms of ℓ_2 distortion. In Fig. 5, we show the RD curves for two test images Lena and Harbor (more experimental results performed on a larger set of test images will be given in our forthcoming paper), which are both of size 512×512. Here the block size $M = 16$. It can be seen that, for bit rates below 1.7 bpp, the proposed method offers better RD performance. The gain in PSNR over [5] can be quite significant for low bit rates. Furthermore, the performance saturation problem (see the curves of Harbor) occurred in [5] can be successfully avoided.

5. CONCLUSIONS

In this paper, we design a novel scalable image coding scheme for stream cipher encrypted images. The base layer compresses a series of non-overlapping patches of the uniformly down-sampled version of the encrypted image. Based on the free side information offered by base layer coding, the enhancement layer strategically selects additional pixel samples to code. The decoder then applies an iterative, multi-scale technique to reconstruct the image from all available samples. Experimental results verify the superior coding performance of our proposed scheme, especially at the low and medium rate regions.

6. REFERENCES

[1] M. Johnson, P. Ishwar, V. M. Prabhakaran, D. Schonberg, and K. Ramchandran,"On compressing encrypted data," *IEEE Trans. on Signal Process.*, vol. 52, no. 10, pp. 2992-3006, Oct. 2004.

[2] R. Lazzeretti and M. Barni,"Lossless compression of encrypted grey-level and color images," in *Proc.16th Eur. Signal Processing Conf. (EUSIPCO 2008)*, Lausanne, Switzerland, 2008.

[3] A. Kumar and A. Makur,"Distributed source coding based encryption and lossless compression of gray scale and color images," in *Proc. MMSP*, pp. 760-764, 2008.

[4] W. Liu, W.J. Zeng, L. Dong, and Q.M. Yao,"Efficient compression of encrypted grayscale images," *IEEE Trans. on Imag. Process.*, vol. 19, no. 4, pp. 1097–1102, April 2010.

[5] X. Zhang, G. Feng, Y. Ren, and Z. Qian, "Scalable coding of encrypted images," *IEEE Trans. on Imag. Process.*, vol. 21, no. 6, pp. 3108–3114, June 2012.

[6] X. Kang, A. Peng, X. Xu, and X. Cao, "Performing scalable lossy compression on pixel encrypted images," *EURASIP J. on Imag. and Video Process.*, 2013:32, 2013.

[7] X. Zhang,"Lossy compression and iterative recobstruction for encrypted image," *IEEE Trans. on Inf. Forensics and Security*, vol. 6, no. 1, pp. 53-58, March 2011.

[8] J. Zhou, X. Liu, O.C. Au, and Y. Tang, "Designing an efficient image encryption-then-compression system via prediction error clustering and random permutation," *IEEE Trans. on Inf. Forensics and Security*, vol. 9, no. 1, pp. 39-50, Jan. 2014.

[9] A.J. Menezes, P.C. Van Oorschot, and S.A. Vanstone *Handbook of Applied Cryptography*, CRC Press, 1997

[10] X. Zhang and X. Wu, "Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation," *IEEE Trans. on Imag. Process.*, vol. 17, no. 6, pp. 887–896, 2008.