Hindawi Publishing Corporation
EURASIP Journal on Information Security
Volume 2008, Article ID 179290, 18 pages
doi:10.1155/2008/179290

Review Article

Overview on Selective Encryption of Image and Video: Challenges and Perspectives

A. Massoudi, F. Lefebvre, C. De Vleeschouwer, B. Macq, and J.-J. Quisquater

Thomson R&D France, Technology Group, Corporate Research, Security Laboratory 1, avenue Belle Fontaine, 35576 Cesson-Sévigné Cedex, France

Correspondence should be addressed to A. Massoudi, ayoub.massoudi@gmail.com

Received 10 January 2008; Accepted 24 November 2008

Recommended by Q. Sun

In traditional image and video content protection schemes, called fully layered, the whole content is first compressed. Then, the compressed bitstream is entirely encrypted using a standard cipher (DES, AES, IDEA, etc.). The specific characteristics of this kind of data (high-transmission rate with limited bandwidth) make standard encryption algorithms inadequate. Another limitation of fully layered systems consists of altering the whole bitstream syntax which may disable some codec functionalities. Selective encryption is a new trend in image and video content protection. It consists of encrypting only a subset of the data. The aim of selective encryption is to reduce the amount of data to encrypt while preserving a sufficient level of security. This computation saving is very desirable especially in constrained communications (real-time networking, high-definition delivery, and mobile communications with limited computational power devices). In addition, selective encryption allows preserving some codec functionalities such as scalability. This tutorial is intended to give an overview on selective encryption algorithms. The theoretical background of selective encryption, potential applications, challenges, and perspectives is presented.

1. INTRODUCTION

Because of the explosion of networks and the huge amount of content transmitted along, securing video content is becoming more and more important. A traditional approach for content access control is to first encode data with a standard compressor and then to perform full encryption of the compressed bitstream with a standard cipher (DES, AES, IDEA, etc.). In this scheme, called *fully layered*, compression and encryption are totally disjoint processes. The media stream is processed as a classical text data with the assumption that all symbols or bits in the plain text are of equal importance. This scheme is relevant when the transmission of the content is unconstrained. In situations where only few resources are available (real-time networking, high-definition delivery, low memory, low power, or computation capabilities), this approach seems inadequate. Shannon [1] pointed out the specific characteristic of image and video content: high-transmission rate and limited allowed bandwidth, which justifies the inadequacy of standard cryptographic techniques for such content. Another limitation of the fully layered scheme consists of altering the original bitstream syntax. Therefore, many functionalities of the encoding scheme may be disabled (e.g., scalability). Some recent works explored a new way of securing the content, named, *partial encryption or selective encryption, soft encryption, perceptual encryption*, by applying encryption to a subset of a bitstream. The main goal of selective encryption is to reduce the amount of data to encrypt while achieving a required level of security. An additional feature of selective encryption is to preserve some functionalities of the original bitstream (e.g., scalability). The general approach is to separate the content into two parts. The first part is the *public part*, it is left unencrypted and made accessible to all users. The second part is the *protected part*; it is encrypted. Only authorized users have access to *protected part*. One important feature in selective encryption is to make the protected part as small as possible.

How to define public and protected parts depends on the target application. In some applications (video on demand, database search, etc.), it could be desirable to encourage customers to buy the content. For this purpose, only a soft visual degradation is achieved, so that an attacker would still understand the content but prefer to pay to access the full-quality unencrypted content. However, for sensitive data (e.g., military images/videos, etc.), hard visual degradation could be desirable to completely disguise the visual content. The peak signal-to-noise ratio (PSNR) is the common criterion used to evaluate visual degradation.

This paper is intended to give an overview of state-of-the-art selective encryption algorithms. We introduce selective encryption in a close link to Shannon's work on information theory in Section 1.2. Evaluation criteria of selective encryption algorithms are presented in Section 1.2. In Section 1.3, we give one classification of selective encryption algorithms. Section 2 proposes potential applications of selective encryption. In Section 3, we will present a summary of different selective encryption algorithms, their advantages, and limitations. In Section 4, based on previous discussion, we will discuss the principal challenges and perspectives for selective encryption.

1.1. Shannon and selective encryption

In [2–4], Lookabaugh pointed out the close link between selective encryption and Shannon's work on communication and security [1]. It is well known that statistics for image and video data differ much from classical text data. Indeed, image and video data are strongly correlated and have strong spatial/temporal redundancy. In addition, contrarily to banking information or other highly sensitive information, the image and video content has high-information rate with low value from the security point of view. Shannon highlighted the relationship between source statistics and the ciphertext security; a secure encryption scheme should remove all the redundancies in the plaintext, so that no exploitable correlation is observed in the ciphertext. Shannon introduced the equivocation function as a measure of how much a cryptanalyst is uncertain of the plaintext observing a set of ciphertexts. Figure 1 illustrates the definition above. A unicity distance n_u is defined as the minimum number of ciphertext blocks required to yield a unique solution in a ciphertext-only attack, this is given by

$$n_u = \frac{H(k)}{r}, \tag{1}$$

where $H(k)$ is the key entropy, and $r$is the plaintext redundancy. From this, we can say that the less redundant the source code is, the more secure the ciphertext is. Shannon favors a fully layered system (see Figure 2), where perfect lossless compression is first performed to remove "all" redundancies from the plaintext (a perfect compressor achieves a rate equal to the source entropy), and then full encryption is applied. Shannon argues that the compressor should be perfect, this means that, given a plaintext P, let P' be its "perfect" compression by the perfect compressor. We can split P' into two parts P'_1 and P'_2. Then, let C_1 and C_2 be the encryption of P'_1 and P'_2 by the encryption algorithm (see Figure 2). Perfect compression implies that if we know only P'_1, then P'_2 is completely unpredictable.

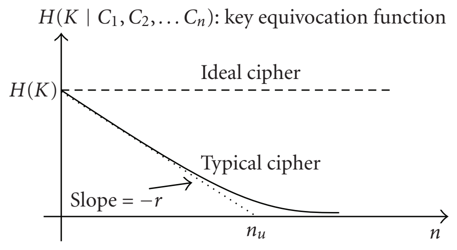

Figure 1: Key equivocation function.

This can be demonstrated using a proof by contradiction. If the statement above was false, then an extra prediction block would yield additional compression of P'_2 based on P'_1. This is impossible since we assumed that the compression is perfect [3]. This result is very interesting; let us consider a configuration, where only a subset of the compressed bitstream requires protection (e.g., P'_1) we can replace the encryption block by a selective encryption one. Only the protected subset is encrypted (P'_1 as illustrated in Figure 3), and the security of the ciphertext is preserved for the same reasons discussed above, with the assumption that all redundancies of the source were removed. P'_1 is protected and unpredictable from P'_2 because the compressor is perfect.

Hence, good compression is a good help for the security of selective encryption. The only question that remains is which part to encrypt to obtain a desired visual degradation. In Shannon's theory, the energy of the "perfectly" compressed plaintext is uniformly distributed, thus encrypting a fraction of the compressed plaintext would yield the same fraction of distortion on the ciphertext. However, most existing compression algorithms are not perfect and concentrate information energy unevenly in the bitstream; for example, in JPEG, the bits that encode the DC coefficients have stronger impact on the reconstruction quality than the AC coefficients. In wavelet-based compression algorithms, most of the signal energy is concentrated in lower resolutions. One advantage of energy concentration is that it gives a hint about which part of the bitstream to encrypt. Most state-of-the-art selective encryption algorithms exploit this energy concentration.

This gap between theoretical selective encryption which is based on perfect compression and existing selective encryption algorithms makes the security aspect more difficult to evaluate. In most cases, visual degradation is used as the exclusive security measure of selective encryption by assuming that harder visual distortion implies more security. It turns out that this argument is not relevant as can be observed in related works.

1.2. Evaluation criteria

We need to define a set of evaluation criteria that will help evaluating and comparing selective encryption algorithms. Some criteria listed below are gathered from the literature. We introduce new criteria that were not considered previously.

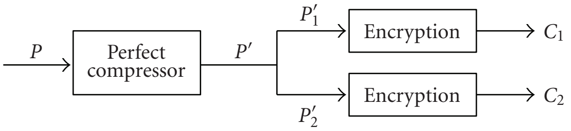

Figure 2: Fully layered system: the whole compressed bitstream is encrypted.

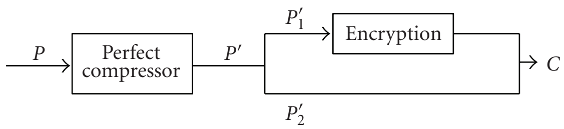

Figure 3: In perfect compression configuration, a subset of the bitstream can be encrypted; protected part is not predictable from the public one.

(I) Tunability (T)

Most of the proposed algorithms in the literature use static definition of encrypted part and encryption parameters. This property limits the usability of the algorithm to a restricted set of applications. It could be very desirable to be able to dynamically define the encrypted part and the encryption parameters with respect to different applications and requirements.

(II) Visual degradation (VD)

This criterion measures the perceptual distortion of the cipher image (or video) with respect to the plain image (or video). It assumes that the cipher image (or video) can be decoded and viewed without decryption. This assumption is not satisfied for all existing algorithms. In some applications, it could be desirable to achieve enough visual degradation, so that an attacker would still understand the content but prefer to pay to access the unencrypted content. However, for sensitive data (e.g., military images/videos), high visual degradation could be desirable to completely disguise the visual content. For this reason, tunability property is very important to be able to tune the visual degradation of the encrypted content depending on the target application and requirements. The peak signal-to-noise ratio (PSNR) is the main metric used in the literature to measure visual degradation. Visual degradation is a subjective criterion that is why it is difficult to define a threshold for acceptable visual distortion regarding a given application.

(III) Cryptographic security (CS)

Most of the research works on selective encryption evaluate the security level based only on visual degradation. In [5], Tang proposes a selective encryption algorithm based on DES encryption of DC coefficients and replacing the zigzag scan of the AC coefficients by a random permutation. The visual degradation achieved is very high, but the cryptographic security of the algorithm is very weak as pointed out in [6, 7]. The cryptographic security should rely on

(i) the encryption key (of a well-scrutinized encryption algorithm),

(ii) unpredictability of the encrypted part.

This criterion will be explained in more detail in Section 4.1.2.

(IV) Encryption ratio (ER)

This criterion measures the ratio between the size of the encrypted part and the whole data size. Encryption ratio has to be minimized by selective encryption.

(V) Compression friendliness (CF)

A selective encryption algorithm is considered compression friendly if it has no or very little impact on data compression efficiency. Some selective encryption algorithms impact data compressibility or introduce additional data that is necessary for decryption. It is desirable that this impact remains limited.

(VI) Format compliance (FC)

The encrypted bitstream should be compliant with the compressor. Any standard decoder should be able to decode the encrypted bitstream without decryption. This property is very important because it allows preserving some features of the compression algorithm used (e.g., scalability).

(VII) Error tolerance (ET)

This criterion is not very considered in the literature. It is very desirable especially in networks prone to errors. As standard ciphers are required to have strong avalanche effect, a single bit error that occurs in the encrypted bitstream during transmission will propagate many other bits after decryption. This causes decoding failure or important distortion to the plain data at the receiver side. A challenge is to design a secure selective encryption algorithm that trades off important avalanche effect and error tolerance.

1.3. Classification of selective encryption algorithms

One possible classification of selective encryption algorithm is relative to when encryption is performed with respect to compression. This classification is adequate since it has intrinsic consequences on selective encryption algorithms behavior. We consider three classes of algorithms as follows.

(I) Precompression

Selective encryption algorithms from this class perform encryption before compression (resp., decompression before decryption) (see Figure 4). Note that these algorithms are inherently format compliant and generally inapplicable

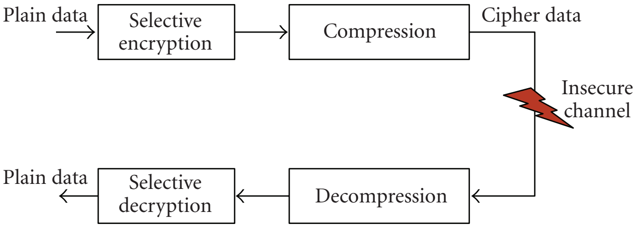

FIGURE 4: Precompression approach.

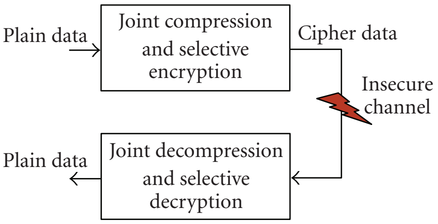
Plain data
Joint compression and selective encryption
Cipher data
Insecure channel
Plain data
Joint decompression and selective decryption

FIGURE 5: Incompression approach.

for lossy compression. Finally, in most cases, performing encryption prior to compression causes bandwidth expansion which adversely impact compression efficiency. Hence, this class of algorithms is generally not compression friendly.

(II) Incompression

Selective encryption algorithms from this class perform joint compression and encryption (resp., joint decompression and decryption) (see Figure 5). Algorithms from this class imply modifications of both encoder and decoder which may adversely impact format compliance and compression friendliness.

(III) Postcompression

Selective encryption algorithms from this class perform compression before encryption (resp., decryption before decompression) (see Figure 6). This class of algorithms is generally compression friendly; small overhead can be introduced to send the encryption key or some information about encryption. Encryption and decryption do not need modifications at encoder or decoder sides. Finally, it was suggested in [8] that postcompression class is inherently nonformat compliant. In this paper, we give example of existing algorithms that achieve format compliance by using pattern-constrained encryption.

2. APPLICATIONS

Digital multimedia content is becoming widely used over networks and public channels (cable, satellite, wireless networks, Internet, etc.), which is unsecured transmission media. Many applications that exploit these channels (pay-TV, videoconferences, medical imaging, etc.) need to rely on access control systems to protect their content. Standard cryptographic techniques can guarantee high level of security but at the cost of expensive implementation and important transmission delays. Selective encryption comes as an alternative that aims at providing sufficient security with an important gain in computational complexity and delays. This allows a variety of possible applications for selective encryption. Below, we give a set of potential applications as follows.

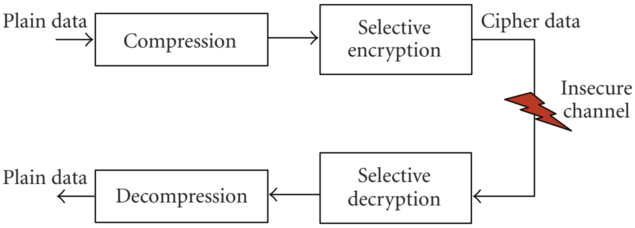

FIGURE 6: Postcompression approach.

(I) Mobile communication

PDAs, mobile phones, and other mobile terminals are more and more used for multimedia communication (voice, image, video, etc.) while still requiring copyright protection and access control. Their moderate resolution, computational power, and limited battery life impose to make an effort in reducing the encryption computational complexity to save battery life, silicon area, and cost. Image and video content have lower value than banking information, for example. Thus, it is not necessary to encrypt the whole data. It would be enough to degrade content quality so that people would prefer to buy a full-quality version.

(II) Monitoring encrypted content

One can imagine a situation where the encrypted content itself is usable for monitoring. For example, in many applications such as military images, video surveillance (where some faces have to be scrambled), media audience, identifying a partially encrypted content without decryption can be desirable.

(III) Multiple encryptions

Efficient overlay of more than one encryption system within a single bitstream can be very desirable. In a scheme where a TV broadcaster using an encryption system that is proprietary of one supplier wants to introduce new encryption systems of new independent suppliers, he would like to optimize bandwidth use by avoiding duplicating every channel on the network. Selective encryption could be very helpful; only a small fraction of the channel is duplicated (the part that will be encrypted). Each duplicated part will go through one supplier equipment and be encrypted by its encryption system. The remaining part (the shared one) will be sent once in the network and in the clear. Sony's *Passage* system proposed for the US cable market is a concrete example of this application [9]. This solution is particularly desirable when the suppliers are not willing to agree on a shared scrambling solution as done in DVB Simulcrypt [10].

(IV) Transcodability/scalability of encrypted content

These are very desirable properties in image and video communication. Some compression algorithms such as JPEG-2000 allow natural transcodability/scalability thanks to its embedded-code nature. For some other algorithms it is necessary to decompress and recompress at lower bitrate at intermediate routers of the transmission channel. When the content is fully encrypted, decryption, decompression, and recompression at lower bitrate and reencryption are needed at intermediate routers. It may also cause important transmission delays and defeat the security of the system since access to the encryption key is needed at the network nodes. Selective encryption could be a good response to this problem. Encrypting a small fraction of the content while sending the remainder in the clear allows transcodability and scalability without accessing the encryption keys; the basic part (needed by all users) is sent in the clear (unencrypted) while the encrypted enhancement part is sent only to authorized users who paid to access the full-quality content.

(V) Database search

Selectively encrypted content can be used as low-quality previews that are made public. This preview will be used as a catalog to select content and pay to be able to decrypt and view it.

(VI) Renewable security systems

In their eternal battle against pirates, digital rights management systems have to periodically update their technologies and equipments all along the network. Changing the whole infrastructure would be very costly. Selective encryption can avoid the burden of having to change a whole system. Because of computational complexity saving due to selective encryption, it is possible to move to software solutions which are less expensive and can be easily and economically updated.

3. RELATED WORK

3.1. Precompression

Tang, 1996. The basic idea of the selective encryption algorithm proposed in [5] is to selectively encrypt I-frames of the MPEG stream; DES on DC coefficients (preferably in CBC mode to avoid dictionary attack) and random permutation on the AC coefficients instead of the standard zigzag. This is done before compression.

(a) *Tunability:* the algorithm is not tunable since encryption parameters are static.

(b) *Visual degradation:* since intraframes are very important in MPEG compression (all B- and P-frames are computed accordingly to I-frames), by encrypting them, high-visual degradation is achieved.

(c) *Cryptographic security:* the AC coefficients zigzag scan used in I-frames encoding is replaced by a pseudorandom permutation. Statistics of the AC coefficients are preserved. Therefore, ciphertext-only, chosen, and known-plaintext attacks are feasible and allow recovering all AC coefficients. Qiao et al. [6] and Uehara and Safavi-Naini [7] propose cryptanalytic attacks (chosen-plaintext attacks) on this approach. The DC coefficient can be set to a fixed value while still having a comprehensible result, and then a chosen or known-plaintext attack can be conducted to reconstruct the AC coefficients and get a semantically good reconstruction [11]. Two conclusions can be made. First, energy concentration is not systematically a good criterion for selective encryption. Second, high-visual distortion does not mean high security level.

(d) *Encryption ratio:* not specified.

(e) *Compression friendliness:* the nonoptimal scanning of the DCT coefficients introduces loss in compression efficiency of about 40% [6]. Indeed, this adversely affects Huffman encoding (due to distortion of the probability distribution of run-lengths for AC coefficients).

(f) *Format compliance:* the proposed scheme is compliant to JPEG and MPEG standards.

(g) *Error tolerance:* the proposed algorithm is not tolerant to errors that occur at DC coefficients. The avalanche effect of DES in CBC mode causes important error propagation.

(h) *Data type:* image and video.

Shi and Bhargava, 1998. In [12], the authors proposed video encryption algorithm (VEA) which uses a secret key to randomly change the signs of all DCT coefficients in an MPEG stream (this is justified by the fact that DCT sign bits are very random, thus neither predictable nor compressible). In [13], the authors present a new version of VEA reducing computational complexity; it consists in encrypting the sign bits of differential values of DC coefficients of I-frames and sign bits of differential values of motion vectors of B- and P-frames.

(a) *Tunability:* not tunable, the proposed algorithm relies on static parameters.

(b) *Visual degradation:* high-visual degradation due to the encryption of DCT coefficients and motion vectors.

(c) *Cryptographic security:* the first version of VEA [12] is only secure if the secret key is used once. Otherwise, knowing one plaintext and the corresponding ciphertext, the secret key can be computed by

XORing the DCT sign bits. Both versions of VEA are vulnerable to chosen plaintext attacks; in [12], it is feasible to create a repetitive/periodic pattern and then compute its inverse DCT. The encryption of the image obtained will allow us to get the key length and even compute the secret key by chosen-plaintext attack.

(d) *Encryption ratio:* not specified.

(e) *Compression friendliness:* not specified.

(f) *Format compliance:* the encrypted bitstream is MPEG compliant.

(g) *Error tolerance:* any error in motion vector bits may have important adverse impact on the decidability of the bitstream.

(h) Data type: video.

Shi, Wang and Bhargava, 1999. In [14], a new version of the modified VEA presented in [13] is proposed, called real-time video encryption algorithm for (RVEA). It encrypts selected sign bits of the DC coefficients and/or sign bits of motion vectors using DES or IDEA. Sixty four sign bits are encrypted per frame (starting by DC coefficients because they concentrate most of the frame energy).

(a) *Tunability:* not tunable.

(b) *Visual degradation:* changing the sign bit of one DC coefficient will affect all the following ones in I-frames (since they are differentially encoded), the same thing applies for motion vectors in P- and B-frames; the sign changes not only the direction but also motion magnitude, since they are differentially encoded. The visual degradation achieved is very high.

(c) *Cryptographic security:* bounding the encryption to the first 64 sign bits is not sufficient from the security point of view. Indeed, when considering high-resolution videos with high bitrate, the first 64 bits represent a very small fraction of the data.

(d) *Encryption ratio:* only 64 bits are encrypted per frame. Thus, encryption reduction depends on the image bitrate.

(e) *Compression friendliness:* not specified.

(f) *Format compliance:* the proposed scheme is MPEG compliant.

(g) *Error tolerance:* poor error tolerance is achieved due to motion information encryption.

(h) *Data type:* video.

Podesser, Schmidt and Uhl, 2002. In [15], a selective bitplane encryption (using AES) is proposed, several experiments were conducted on 8-bit grayscale images, and the main results retained are the following: (1) encrypting only the MSB is not secure; a replacement attack is possible [15], (2) encrypting the first two MSBs gives hard visual degradation, and (3) encrypting three bitplanes gives very hard visual degradation.

(a) *Tunability:* the algorithm is not tunable; a fixed number of bits need to be encrypted to guarantee confidentiality.

(b) *Visual degradation:* for 8 bits per pixel uncompressed image, hard visual degradation (of 9 dB) can be observed for a minimum of 3 MSB bits encrypted.

(c) *Cryptographic security:* even when a secure cipher is used (AES), the selective encryption algorithm proposed is vulnerable to replacement attacks [15]. This attack does not break AES but replaces the encrypted data with an intelligible one. It is worth to note that visual distortion is a subjective criterion and does not allow to measure security as illustrated in this example.

(d) *Encryption ratio:* at least 3 bitplanes over 8 (more than 37.5%) of the bitstream have to be encrypted using AES to achieve sufficient security.

(e) *Compression friendliness:* this algorithm is intended for uncompressed data. However, important bandwidth expansion is introduced by selectively encrypting MSBs which adversely impact the compressibility of encrypted images.

(f) *Format compliance:* as a precompression algorithm, it is format compliant.

(g) *Error tolerance:* the avalanche effect of AES causes important error propagation.

(h) *Data type:* uncompressed image.

Zeng and Lei, 2003. In [16], selective encryption in the frequency domain (8 × 8 DCT and wavelet domains) is proposed. The general scheme consists of selective scrambling of coefficients by using different primitives (selective bit scrambling, block shuffling, and/or rotation).

(I) Wavelet transform case

The proposed scheme combines two primitives.

(i) *Selective bit scrambling:* it is a bitplane selective encryption; each individual coefficient bitplane is partitioned into a sign bit, which is very random and uncorrelated with neighboring coefficient sign bits, thus highly unpredictable. Then significance bits (the first nonzero magnitude bit and all subsequent zero bits if any), these give a range for the coefficient value. These bits have low entropy and thus are highly compressible. Finally, the refinement bits (all remaining bits) are uncorrelated with neighboring coefficients and are randomly distributed.The authors propose to randomly scramble sign bits and refinement bits. The encryption algorithm is not specified.

(ii) *Block shuffling:* the basic idea is to shuffle the arrangement of coefficients within a block in a way to preserve some spatial correlation; this can achieve sufficient security without compromising compression efficiency. Each subband is split into equal-sized blocks (the block size can be different for each subband). Within the same subband, block coefficients are shuffled according to a shuffling table generated using a secret key (this table can be different from a subband to another or from one frame to another). Since the shuffling is block based, it is expected that most 2D local subband statistics are preserved and compression not greatly impacted.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* high-visual degradation is achieved. Indeed, coefficient change at low resolutions propagates to larger parts at higher resolutions.

(c) *Cryptographic security:* attacking the lowest pyramid level of the wavelet decomposition is much simpler (small block size and high energy concentration) this helps to construct the subsequent levels by correlation.

(d) *Encryption ratio:* about 20% of the data has to be encrypted.

(e) *Compression friendliness:* little impact on compression efficiency is observed (less than 5%).

(f) *Format compliance:* the algorithm proposed is fully compliant to DWT-based compression since the encryption is performed in the transform domain prior to compression.

(g) *Error tolerance:* depends on the encryption algorithm used to scramble sign bits.

(h) *Data type:* image and video.

(II) DCT transform case

The 8 × 8 DCT coefficients can be considered as individual local frequency components located at some subband. The same scrambling operations as described above (block shuffling and sign bits change) can be applied on these “subbands.” I-, B-, and P-frames are processed in different manners. For I-frames, the image is first split into segments of macroblocks (e.g., a segment can be a slice), blocks/macroblocks of a segment can be spatially disjoint and chosen at random spatial positions within the frame. Within each segment, DCT coefficients at the same frequency location are shuffled together (in order to preserve coefficients distribution property). Then, sign bits of AC coefficients are randomly changed and DC coefficients (which are always positive for intracoded blocks) are flipped with respective threshold (e.g., $255*8/2$ = maximum DC value/2). There may be many intracoded blocks in P- and B-frames. At least DCT coefficients of the same intracoded block in P- or B-frames are shuffled. Sign bits of motion vectors are also scrambled.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* high-visual degradation is achieved. Indeed, most of the image energy is concentrated in DC coefficients, thus, encrypting them affects considerably the image content.

(c) *Cryptographic security:* vulnerable to chosen and known plaintext attacks since it is based only on permutations. In addition, replacing the DC coefficients with a fixed value still gives an intelligible version of the image.

(d) *Encryption ratio:* if we consider only the AC sign bit encryption, it represents 16 to 20% of data. This is relatively high [16].

(e) *Compression friendliness:* a bitrate increase by about 20% is observed.

(f) *Format compliance:* compliant with JPEG and MPEG standards.

(g) *Error tolerance:* depends on the encryption algorithm used to scramble sign bits.

(h) *Data type:* image and video.

Van de Ville, Philips, Van de Walle, and Lemahieu, 2004. A particular orthonormal transform is used in this proposal, the discrete prolate spheroidal sequences (DPSSs) [17]. This is an adapted base to represent band limited signals (which is the case for 2D images). A bandwidth preserving scrambling is proposed; the image signal is projected on the DPSS (which is a base for band limited signals). Then, the transform coefficients are scrambled using an orthonormal (thus energy preserving) transform.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* depends on the number of coefficients to scramble.

(c) *Cryptographic security:* a large key space is obtained due to the use of equivalent Hadamard matrices in the scrambling. However, statistical correlations exist between coefficients to encrypt; this leakage has been exploited to mount an error-concealment-based attack (ECA) [18]. Finally, the Hadamard matrix-based encryption has insufficient diffusion, this leads to a reduction in key space. Experimental results show that when guessing 100 random keys, the best recovered image has low-visual degradation compared to the unencrypted one.

(d) *Encryption ratio:* variable, it depends on the number of coefficients to scramble.

(e) *Compression friendliness:* limited bandwidth expansion is allowed by this proposal. However, the major drawback of this scheme is that the encryption is lossy. Indeed, the encryption process implies a rounding operation that induces precision loss (so inadequate to lossless compression).

(f) *Format compliance:* as a precompression algorithm, it is format compliant.

(g) *Error tolerance:* important error propagation due to the avalanche property of Hadamard matrices used in encryption.

(h) *Data type:* image.

3.2. In-compression

Meyer and Gadegast, 1995. The algorithm is proposed for MPEG selective encryption (called SECMPEG). It modifies the MPEG stream [19]. It uses RSA or DES (in CBC mode) and implements 4 levels of security.

(i) Encrypting all stream headers.

(ii) Encrypting all stream headers and all DC and lower AC coefficients of intracoded blocks.

(iii) Encrypting I-frames and all I-blocks in P- and B-frames.

(iv) Encrypting all the bitstreams.

(a) *Tunability:* the algorithm can be considered as tunable since many security levels are allowed.

(b) *Visual degradation:* the encrypted content is not MPEG compliant, and thus cannot be viewed without decryption.

(c) *Cryptographic security:* many security levels can be obtained. Encrypting only stream headers is not sufficient since this part is easily predictable.

(d) *Encryption ratio:* the number of I blocks in P or B frames can be of the same order as the number of I blocks in I frames. This reduces considerably the efficiency of the selective encryption scheme [20].

(e) *Compression friendliness:* no impact is observed on the compression efficiency.

(f) *Format compliance:* the encoder proposed is not MPEG compliant since it requires major additions and changes to the standard; a special encoder/decoder is required to read unencrypted SECMPEG streams.

(g) *Error tolerance:* the ciphers used for encryption have important avalanche properties, especially in CBC mode. Hence, poor error tolerance is achieved.

(h) *Data type:* video.

Wu and Kuo, 2001. In [11, 21], based on a set of observations, the authors point out that energy concentration does not mean intelligibility concentration. Indeed, they discussed the technique proposed by Tang [5]. They show that by fixing DC values at a fixed value and recovering AC coefficients (by known or chosen plaintext attacks), a semantically good reconstruction of the image is obtained. Even using a very small fraction of the AC coefficients does not fully destroy the image semantic content. The authors argued that both orthogonal transform-based compression algorithms followed by quantization and compression algorithms that end with an entropy coder stage are bad candidates to selective encryption. They investigate another approach that turns entropy coders into ciphers. They propose two schemes for the most popular entropy coders: multiple Huffman tables (MHTs) for the Huffman coder and multiple state index (MSI) for the QM arithmetic coder.

(I) MHT

The authors propose a method using multiple Huffman coding tables. Four Huffman tables are published, and millions of different tables are generated using a technique called Huffman tree mutation [11, 21].

(a) *Tunability:* not tunable.

(b) *Visual degradation:* very high-visual degradation can be achieved.

(c) *Cryptographic security:* Gillman and Rivest [22] showed that decoding a Huffman coded bitstream without any knowledge about the Huffman coding tables would be very difficult. However, the basic MHT is vulnerable to known and chosen plaintext attacks as pointed out in [23].

(d) *Encryption ratio:* variable, it depends on the size of the data to encrypt. Indeed, the larger the data is, the smaller the relative size of the Huffman table will be.

(e) *Compression friendliness:* no impact on compression is observed, the encryption does not affect the probability distribution of symbols.

(f) *Format compliance:* not compliant, the decoder needs to decrypt the Huffman table to be able to decompress.

(g) *Error tolerance:* as Huffman coding relies on variable length codes, any single codeword error may propagate at many subsequent codewords.

(h) *Data type:* image and video.

(II) MSI

The arithmetic QM coder is based on an initial state index; the idea is to select 4 published initial state indices and to use them in a random but secret order.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* very high-visual degradation can be achieved.

(c) *Cryptographic security:* high security level. It is very difficult to decode the bitstream without the knowledge of the state index used to initialize the MQ coder.

(d) *Encryption ratio:* very low encryption ratio is achieved. However, the computation cost is relatively high; this is due to multiple updates in the QM coder states.

(e) *Compression friendliness:* a little effect on compression efficiency is observed. This is due to multiple initializations of the QM coder due to initial state index changing.

(f) *Format compliance:* not compliant. It is impossible to decode without the encryption key.

(g) *Error tolerance:* frequent reset of state indices allows high error tolerance.

(h) *Data type:* image and video.

Wen, Severa, Zeng, Luttrel, and Jin, 2002. A general selective encryption approach for fixed and variable length codes (FLC and VLC) is proposed in [24]. FLC and VLC codewords corresponding to important information carrying fields are selected. Then, each codeword in the VLC and FLC (if the FLC code space is not full) table is assigned a fixed length code index, when we want to encrypt the concatenation of some VLC (or FLC) codewords, only the indices are encrypted (using DES). Then the encrypted concatenated indices are mapped back to a different but existing VLC.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* very high-visual degradation can be achieved.

(c) *Cryptographic security:* acceptable security level based on the secrecy of the Huffman table.

(d) *Encryption ratio:* good encryption reduction (<15%).

(e) *Compression friendliness:* the encryption process compromises the compression efficiency. Indeed, some short VLC codewords (which are the most probable/frequent) can be replaced by longer ones. This is antagonistic with the entropy coding idea.

(f) *Format compliance:* the proposed scheme isfully compliant to any compression algorithm that uses VLC or FLC entropy coder.

(g) *Error tolerance:* any error affecting one variable length code may potentially propagate to subsequent codewords.

(h) *Data type:* image and video.

Pommer and Uhl, 2003. The algorithm proposed in [25] is based on AES encryption of the header information of wavelet packet encoding of an image, this header specifies the subband tree structure.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* the encrypted content cannot be viewed without decryption.

(c) *Cryptographic security:* no secure against chosen plaintext attack. Because statistical properties of wavelet coefficients are preserved by the encryption, then the approximation subband can be reconstructed. This will give the attacker the size of the approximation subband (lower resolution) and then neighboring subbands can be reconstructed since close subbands contain highly correlated coefficients.

(d) *Encryption ratio:* the encrypted part represents a very small fraction of the bitstream.

(e) *Compression friendliness:* the subband tree is pseudorandomly generated. This adversely impacts the compression efficiency.

(f) *Format compliance:* no format compliant; the encoder does not use standard wavelet packet decomposition.

(g) *Error tolerance:* the avalanche effect of AES cipher causes poor error tolerance.

(h) *Data type:* image.

Lian, Sun, and Wang, 2004. A selective encryption algorithm is proposed for JPEG2000 standard [26]. A quality factor controls the strength of the encryption algorithm. The encryption algorithm is performed in a bottom-up order where detail data (high-resolution coefficients) are encrypted first. The algorithm consists in three steps.

(I) Selective sign bit encryption

A selected number (s) of sign bits are encrypted using a chaotic stream cipher. The quality factor tunes s.

(II) Intra-bitplane permutation

For each bitplane, in each code block, a pseudorandom space filling curve (PR-SFC) is used to permute bits of the same bitplane. It seems that the algorithm uses the same SFC for all bitplanes in a given bitplane. Hence, it is a simple coefficient permutation; this is not secure against ciphertext-only, chosen- and known-plaintext attacks [27, 28]. Each 4 bits of a stripe column are grouped together to form a unit element for the permutation (to be compliant to the JPEG2000 standard). The SFC is chosen to preserve spatial correlation of DWT coefficients. The quality factor p tunes the number of code-blocks to be intra-permuted.

(III) Interblocks permutation

Code blocks within the same subband are permuted using a particular 2D chaotic map, the Cat map. If the quality factor is above a certain threshold, no intercodeblock permutation is performed.

(a) *Tunability:* dynamic encryption parameters can be fine tuned to control visual distortion.

(b) *Visual degradation:* the encryption strength (and hence the visual degradation) can be fine tuned using a quality factor.

(c) *Cryptographic security:* low diffusion effect, the ciphertext is not key sensitive enough. In addition, SFC is vulnerable to ciphertext-only, chosen- and known-plaintext attacks [27, 28].

(d) *Encryption ratio:* variable, it depends on the parameters selected for encryption.

(e) *Compression friendliness:* because bitplane encoding depends from the previous bitplanes encoding, independently encrypting each bitplane of a codeblock will inevitably impact the arithmetic coder compression performance.

(f) *Format compliance:* JPEG2000 compliant.

(g) *Error tolerance:* chaotic stream ciphers allow high error tolerance since each sign bit is independently scrambled by a XOR.

(h) *Data type:* image and video.

Grangetto, Magli, and Olmo, 2006. The basic approach proposed in [29] is a randomization of the arithmetic coder.

This is achieved by randomly swapping the most probable symbol (MSP) and least probable symbol (LSP) intervals. Since only the interval magnitude is important for encoding, the compression performance remains unchanged. Both total and selective encryptions are possible by choosing the layers or resolution levels to encrypt. Selective region encryption is made possible since JPEG2000 is a codeblock-based algorithm. To encrypt a region of interest, we have to apply the encryption on the codeblocks contributing to precincts of the region considered.

(a) *Tunability:* selective to full encryption is allowed. Selective region encryption is allowed with dynamic selection of codeblocks to encrypt.

(b) *Visual degradation:* depends on the number of codeblocks to be encrypted.

(c) *Cryptographic security:* low security, brute force attack is feasible. Indeed, trying 30 millions random keys will allow retrieving the secret encryption key.

(d) *Encryption ratio:* variable, depends on the number of codeblocks to be encrypted.

(e) *Compression friendliness:* no impact on compression.

(f) *Format compliance:* fully compliant to JPEG2000.

(g) *Error tolerance:* since arithmetic coding is context based, any error will propagate to subsequent contexts and adversely impact probabilities computations.

(h) *Data type:* image and video.

Bergeron and Lamy-Bergot, 2005. A syntax compliant encryption algorithm is proposed for H.264/AVC [30]. Encryption is inserted within the encoder. To achieve syntax compliance, selected compliant codewords are randomly permuted with other compliant codewords. The shift used for permutation is determined by the AES counter.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* 25 to 30 dB PSNR drop is achieved. However, blocks at the border of video frames cannot be encrypted. This leakage could be important in some applications.

(c) *Cryptographic security:* the main drawback of this scheme is the lack of cryptographic security. Indeed, the security of the encrypted bitstream does not depend more on the AES cipher. It depends on the size of the compliant codewords. Hence, the diffusion of the AES cipher is reduced to the plaintext space size. In addition, a bias is introduced in the ciphertext. This bias depends on the key size and the plaintext space size.

(d) *Encryption ratio:* the paper does not give precise values for overall encryption ratio. However, it is mentioned that about 25% of I-slices and 10–15% of P-slices are encrypted. Since intracoded slices can represent 30–60%, the encryption ratio is expected to be relatively high.

(e) *Compression friendliness:* negligible overhead is introduced (0.1%) by the insertion of encryption key.

(f) *Format compliance:* the encrypted bitstream is decodable by any standard decoder without decryption. However, for decryption, a modified decoder is required.

(g) *Error tolerance:* the randomness of the permutation causes poor error tolerance. Indeed, one single bit error could result in many bit errors if the new permuted codewords have many different bits.

(h) *Data type:* video.

Engel and Uhl, 2006. In [31], a JPEG2000 lightweight encryption scheme is proposed. Only lower resolutions are compressed with classical dyadic wavelet transform. For higher resolutions, the algorithm relies on a secret transform domain constructed with anisotropic wavelet packets (AWPs). The aim of this proposal is to allow transparent encryption for applications requiring low-resolution preview. Therefore, low resolution is accessible by all users and decodable with any JPEG2000 compliant codec.

(a) *Tunability:* limited tunability is permitted. Only lightweight encryption is allowed. Indeed, this algorithm does not allow encrypting lower resolutions. It is intended to particular applications with public thumbnail preview.

(b) *Visual degradation:* high-visual degradation is achievable.

(c) *Cryptographic security:* encryption key space is very large ensuring high security level.

(d) *Encryption ratio:* very low, only the subband tree structure is kept secret.

(e) *Compression friendliness:* only a slight drop in compression performance can be observed.

(f) *Format compliance:* no compliant to JPEG2000, the encrypted bitstream is not decodable without the secret wavelet transform.

(g) *Error tolerance:* it offers poor error tolerance since any error in the encrypted parameters for generating random AWP would severely impact the decoding of the bitstream.

(h) *Data type:* image and video.

3.3. Postcompression

Spanos and Maples, 1995. Aegis mechanism is proposed [32]; it consists in DES (CBC mode) encryption of intraframes, video stream header (all the decoding initialization parameters: frame size, frame rate, bitrate, etc.), and the ISO 32 bits end code of the MPEG stream. Experimental results were conducted by the authors showing the importance of selective encryption in high bitrate video transmission to achieve acceptable end-to-end delay. It is also shown that full encryption creates bottleneck (important end-to-end delay and overflow in buffers) in high bitrate distributed video applications.

(a) *Tunability:* no tunability is allowed.

(b) *Visual degradation:* the encrypted content is not MPEG compliant, and thus cannot be viewed without decryption.

(c) *Cryptographic security:* Agi and Gong [33] showed that this algorithm has low security since encrypting of only I-frames offers limited security because of the intercorrelation of frames; some blocks are intracoded in P- and B-frames. Furthermore, P- and B-frames are highly correlated when they correspond to the same I-frame. They also underlined that it is unwise to encrypt stream headers since they are predictable and can be broken by plaintext-ciphertext pairs. Alattar and Al-Regib [34], apparently unaware of Agi and Gong work [33], stressed the same security leakage.

(d) *Encryption ratio:* I-frames alone occupy about 30 to 60% of the whole video stream, which is quite high. Thus, no important encryption saving is achieved. It is suggested that reducing I-frames frequency could achieve better encryption efficiency; on the other hand, this will adversely impact compression performance and random acquisition delay in case of channel change.

(e) *Compression friendliness:* the encryption is performed after compression, thus no impact is observed on the compression efficiency.

(f) *Format compliance:* the resulting bitstream is not MPEG compliant; encrypting the end code conceals the MPEG syntax.

(g) *Error tolerance:* DES in CBC mode offers poor error tolerance due to its avalanche property.

(h) *Data type:* video.

Alattar and Al-Regib, 1999. In [34], the security of Spanos and Maples algorithm is evaluated [32]. It is argued that motion information has to be disguised when motion information is very important to protect (e.g., military). Spanos and Maples algorithm [32] reveals motion information especially when many blocks are intracoded in P- and B-frames. The proposed technique is an enhancement and improvement to the method proposed in [32]. It requires the transmission of additional information. The proposed scheme consists in the following.

(i) Take all I-blocks and parse the obtained stream into 64-bit segments, encrypt all of them using DES if the last segment is less than 64-bits then leave it unencrypted.

(ii) For predicted blocks in P- and B-frames.

(iii) Group all predicted block headers in one header sub-bitstream.

(iv) Group all prediction block data in one data sub-bitstream.

(v) Parse the header sub-bitstream into 64-bit segments and DES encrypt them.

(vi) Concatenate the encrypted header sub-bitstream with the data sub-bitstream.

(vii) To allow decoding, the length of the header sub-bitstream is transmitted in each slice (in the user section of each slice), this introduces a slight overhead.

(a) *Tunability:* no tunability is allowed.

(b) *Visual degradation:* the encrypted content is not MPEG compliant, and thus cannot be viewed without decryption.

(c) *Cryptographic security:* the algorithm can be considered as secure enough.

(d) *Encryption ratio:* high encryption ratio is required (intracoded blocks represent 30% to 60% of the bitstream).

(e) *Compression friendliness:* a slight overhead is introduced to indicate the header sub-bitstream length.

(f) *Format compliance:* no MPEG compliant; a parser module has to be implemented to interface the encryption/decryption system with the MPEG-1 encoder/decoder.

(g) *Error tolerance:* poor error tolerance is achieved due to avalanche property of DES cipher.

(h) *Data type:* video.

Cheng and Li, 2000. In [35], selective encryption is proposed for quadtree compression algorithm. The compressor output is partitioned into two parts; an “important part” that consists of the quadtree structure, and an “unimportant part” that consists of the leaf values. No encryption algorithm is specified, only the important part is encrypted.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* high-visual degradation can be achieved only for images with high information rate (many colors, details, etc.). But quadtree compression is more efficient at low bitrates (for images with low information).

(c) *Cryptographic security:* no encryption algorithm is specified in [35]. Independently from the encryption algorithm used, brute force attack is practical for low information images where quadtree structure is very simple.

(d) *Encryption ratio:* low encryption ratio is required for typical images with low information content, about 14%. For high bitrate image, the encrypted part can reach about 50%.

(e) *Compression friendliness:* the encryption is performed after compression, no impact on the compression efficiency is observed.

(f) *Format compliance:* quadtree is not part of any compression standard.

(g) *Error tolerance:* depends on the encryption primitive used to encrypt quadtree structure.

(h) *Data type:* image.

Cheng and Li, 2000. The wavelet-based compression algorithm SPIHT partitions the data into two parts [35]. The first part can be considered as the "important part," it consists of significant information (of coefficients and sets) for the two highest levels of the pyramid and the initial threshold parameter n of significance computation (T^n). The second part is the "unimportant part," it consists of sign bits and refinement bits. No encryption algorithm is specified, only the important part is encrypted.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* the algorithm is not format compliant and therefore encrypted content cannot be viewed without decryption key.

(c) *Cryptographic security:* if the two highest resolutions are very small, brute force attack becomes possible to guess the initial threshold and significance information.

(d) *Encryption ratio:* due to the energy concentration obtained by the DWT, only 7% of the bitstream is encrypted.

(e) *Compression friendliness:* no impact on compression efficiency.

(f) *Format compliance:* SPIHT is not part of any compression standard. In addition, since SPIHT algorithm is context based, no decoding/processing is possible without the knowledge of the first significance bits.

(g) *Error tolerance:* poor error tolerance is achieved due to the context nature of SPIHT.

(h) *Data type:* image and video.

Droogenbroeck and Benedett, 2002. The JPEG Huffman coder terminates runs of zeros with codewords/symbols in order to approach the entropy. Appended bits are added to these codewords to fully specify the magnitudes and signs of nonzero coefficients, only these appended bits are encrypted (using DES or IDEA) [36].

(a) *Tunability:* not tunable.

(b) *Visual degradation:* high-visual degradation is achievable.

(c) *Cryptographic security:* about 92% of the data is encrypted using well-scrutinized symmetric ciphers. It would be very difficult to break the encryption algorithm or try to predict the encrypted part.

(d) *Encryption ratio:* very high encryption ratio is required (about 92%).

(e) *Compression friendliness:* the encryption is separated from the Huffman coder and has no impact on the compression efficiency.

(f) *Format compliance:* JPEG compliant.

(g) *Error tolerance:* poor error tolerance is achieved due to avalanche property of symmetric ciphers used.

(h) *Data type:* image.

Sadourny and Conan, 2003. In [37], a signaling scheme is proposed for JPSEC [38]. JPSEC is Part 8 of JPEG2000, called also secure JPEG2000. An important effort has been made in JPSEC to provide a standardized framework to implement security tools and services such as selective encryption, authentication, integrity, and so on. In [37], the signaling scheme proposed is intended to support selective encryption in JPSEC. Two marker segments are used, security components description (SCD) to signal the presence of protected parts in the bitstream and associated encryption parameters and codestream security information (CSI) to signal each individual protected part encryption parameters such as the protection method, some integrity data (hash values, signatures, etc.).

(a) *Tunability:* high flexibility is allowed by the signaling information to encrypt different parts with different encryption parameters.

(b) *Visual degradation:* the tenability of the scheme allows tunable visual degradation.

(c) *Cryptographic security:* depends on encryption parameters.

(d) *Encryption ratio:* depends on encryption parameters.

(e) *Compression friendliness:* the paper presents few overhead tests on encrypted data. A single set of encryption parameters is tested yielding a signaling overhead of 104 bytes. The size of the overhead needs to be measured with respect to image file size and with different encryption parameters.

(f) *Format compliance:* JPEG2000 and JPSEC compliant.

(g) *Error tolerance:* depends on encryption parameters, for example, in the experiments presented in [37], DES is used in CFB mode for encryption which yields poor error tolerance due to chaining mode.

(h) *Data type:* image.

Wu and Deng, 2004. The proposed encryption scheme [39] is a JPEG2000 compliant algorithm which iteratively encrypts codeblock contribution to packets (CCPs). The encryption process acts on CCPs (in the packet data) using stream ciphers or block ciphers. The described proposal is based mainly on stream ciphers with arithmetic module addition. The key stream is generated using RC4. Each CCP is iteratively encrypted until it has no forbidden codewords (in the range [0XFF90, 0XFFFF]) because this range is reserved for packet headers and is necessary for error resiliency and resynchronization.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* depends on the number of CCPs encrypted.

(c) *Cryptographic security:* iterative encryption of CCPs may give a hint for side channel attacks (e.g., timing attack).

(d) *Encryption ratio:* depends on the number of CCPs encrypted. However, the number of iterations per CCP increases exponentially with CCP length [40] which increases the overall effective encryption ratio.

(e) *Compression friendliness:* no impact on compression.

(f) *Format compliance:* fully compliant to JPEG2000 bitstream and preserving scalability and error resiliency which are desirable properties in JPEG2000.

(g) *Error tolerance:* the use of RC4 causes important error propagation.

(h) *Data type:* image and video.

Norcen and Uhl, 2003. JPEG2000 is an embedded bitstream. In addition, most important data is sent at the beginning of the bitstream. Based on these observations, the proposed scheme consists in AES encryption of selected packet data [41]. The algorithm uses two optional markers start of packet (SOP) marker 0xFF91 and end of packet (EPH) marker 0xFF92 to identify packet data. Then, this packet data is encrypted using AES. CFB mode is used because the packet data has variable length. The experiments have been conducted on two kinds of images (lossy and lossless compressed), with different progression orders (resolution and layer progression orders). The evaluation criterion was the visual degradation obtained for a given amount of encrypted data. It was found that for the lossy compressed images, layer progression gives better results. For lossless compressed images, resolution progression gives better results.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* high-visual degradation is achievable by encrypting 20% of the data.

(c) *Cryptographic security:* visual degradation is not the unique criterion that characterizes the security of the algorithm. In [5], the visual degradation achieved is very high while the algorithm security is very weak.

(d) *Encryption ratio:* 20% of the data is encrypted to achieve an acceptable level of visual degradation. However, only resolution and layer progressions are considered.

(e) *Compression friendliness:* no impact on compression.

(f) *Format compliance:* not JPEG2000 compliant. Indeed, forbidden codewords in the range [0XFF90; 0XFFFF] can be generated by the AES-CFB mode.

(g) *Error tolerance:* AES in CFB mode has poor error tolerance.

(h) *Data type:* image and video.

Stütz and Uhl, 2006. In [40], the algorithm proposed by Wu and Deng [39] is revisited. The complexity of the iterative encryption of CCPs was less than estimated in [39], Stütz and Uhl gave a more exact formulation of the CCPs distribution and hence for the encryption complexity [40]. The number of rounds needed to achieve compliant codestream increases exponentially with the CCPs length. In addition, experimental results were conducted to test the practicality of both CCPs and packets iterative encryption. CCPs iterative encryption can be well performing if the compression parameters are well selected (use of sufficient quality layers and/or small precincts with small codeblocks). On the other hand, reducing codeblocks size severely impact compression performance. For packets iterative encryption, it was shown that the distribution of packets length make it impractical. This shows that Wu and Deng approach is not general for JPEG2000 compressed images and special care has to be taken when selecting compression parameters.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* depends on the number of CCPs encrypted.

(c) *Cryptographic security:* iterative encryption of CCPs may give a hint for side channel attacks (e.g., timing attack).

(d) *Encryption ratio:* depends on the number of CCPs encrypted.

(e) *Compression friendliness:* small codeblocks adversely impact compression performance; the MQ coder performs better on large codeblocks. In addition for small packets, the packet headers and marker sequences (e.g., SOP and EPH) will represent an important fraction of the bitstream.

(f) *Format compliance:* JPEG2000 compliant, but the proposed technique is not applicable for any set of compression parameters (many quality layers are needed and small codeblocks).

(g) *Error tolerance:* the use of RC4 causes important error propagation.

(h) *Data type:* image and video.

Engel, Stütz, and Uhl, 2007. In [42], a syntax-compliant encryption method is proposed for JPEG2000. Each codeblock CCP or segment is independently encrypted. The method is based on a new format compliant encryption called ciphertext switching encryption (CSE). A stream cipher is used for encryption with backward checking, each time a forbidden codeword is generated by encryption, it is switched back to plaintext and neighboring codewords are checked back for compliance. This process is iterated until no forbidden codeword is found.

(a) *Tunability:* not tunable.

(b) *Visual degradation:* high-visual degradation can be achieved.

(c) *Cryptographic security:* each time a forbidden codeword is generated, it is switched back to plaintext. In addition, switching impacts all previously encrypted bytes, backward check is necessary for each switched byte. The number of bytes sent in plaintext can be unpredictable.

(d) *Encryption ratio:* depends on the number of CCPs encrypted. However, important memory is required to buffer the previously encrypted bytes for backward check. This check is performed after each byte encryption.

(e) *Compression friendliness:* one major advantage of this scheme is its compression friendliness. Indeed, a negligible overhead of 11 bytes is introduced. Only a short global IV (initial value) is inserted in the bitstream main header. This global IV is used in generating IVs for independent CCPs encryption.

(f) *Format compliance:* JPEG2000 compliant with fine granularity scalability.

(g) *Error tolerance:* the main drawback of this scheme is the need to do backward checking and switching if necessary. Indeed, a single byte error could impact the whole CCP decryption due to the dependency between bytes encryption.

(h) *Data type:* image and video.

Table 1 summarizes the related work with respect to each criteria described above. The main symbols used are

(i) "+" for satisfied criterion,

(ii) "−" for nonsatisfied criterion,

(iii) "H" for high,

(iv) "V" for variable, it is appreciated that visual degradation is variable in order to adapt to different application requirements,

(v) "?" for nonspecified.

It is desirable that visual degradation is variable and dynamically tunable to adapt to different application requirements. Encryption ratio needs to be minimized. Grayed boxes indicate unsatisfied criteria.

4. DISCUSSION AND PERSPECTIVES

4.1. Discussion

As we can see from state-of-the-art summary and Table 1, trading off all aforementioned criteria is a crucial task. We can observe that tunability, cryptographic security, and error tolerance are the main unsatisfied criteria. In the following sections, each of these criteria is discussed.

4.1.1. Tunability

Selective encryption algorithms based on static encryption parameters do not allow tunability. Tunability is a desirable property especially for content protection systems targeting different applications with different requirements in terms of security or visual degradation and different devices with different capabilities in terms of memory, computational power, or display capabilities. It is therefore appreciated to design a tunable selective encryption algorithm with dynamic encryption parameters. Signaling information can be inserted within the bitstream in order to indicate the location of encrypted parts and encryption primitives and functionalities that are used.

4.1.2. Cryptographic security

Very few papers have proposed a serious evaluation of the security of selective encryption algorithms. In most cases, visual distortion (measured using the PSNR) is used as the exclusive criterion for such purpose. However, visual degradation remains a subjective measure. In addition, it has been shown that some selective encryption algorithms that yield important visual distortion may have important security leakages [17, 18]. Cryptanalysis of selective encryption algorithms rely on key recovery (if encryption key space is not large enough) or prediction of encrypted part. Hence, cryptographic security should rely on

(i) the encryption key (of a well-scrutinized encryption algorithm);

(ii) unpredictability of the encrypted part.

As shown in Section 1.1, postcompression selective encryption algorithms are more suited for selective encryption from the security point of view. Indeed, compression eliminates data correlation which reduces the predictability of the encrypted part.

Very few works have been reported on the unpredictability of the encrypted part. Security of the selective encryption algorithm depends on how much and which parts of a message we have to encrypt to ensure that brute force on the encryption key space is easier than brute force attack on the plaintext itself. Otherwise, the attacker could bypass encryption and concentrate his effort on predicting the plaintext. It is hard to find an absolute measure for security. Instead, we define indirect measures that could approximate the security of a selective encryption algorithm. Examples of such measures are entropy, unicity distance guesswork, and α-work factor [43]. Entropy, as suggested by Shannon [1], measures the message uncertainty. It defines the message randomness. It is used to calculate unicity distance [1] which is an approximation of the minimum number of ciphertexts needed in a ciphertext-only attack to yield a unique solution. Guesswork, as suggested in [44, 45], measures the expected number of guesses to perform in optimal brute force attack (where the attacker has perfect knowledge about symbols probability distribution) to find the plain message. In [44, 45], the authors showed that it is not possible to find simple bounds for guesswork (and α-work factor) based on entropy. They found that guesswork can be arbitrarily large while entropy tends to zero. In [44], the author considers entropy inappropriate as confidentiality measure in ciphertext attacks. Based on these observations, [43] proposes guesswork as measure for confidentiality of selectively encrypted messages. We investigate the implications of these results on postcompression selective encryption algorithms.

We consider a message M, compressed by a "perfect compressor." M is composed of n symbols. We arbitrarily choose n_e symbols that will be encrypted ($n_e \leq n$), X designates the encrypted part. The remainder of the message is left unencrypted (Figure 7). The encryption ratio is given by

$$\mathrm{ER} = \frac{n_e}{n}. \tag{2}$$

Table 1: Summary of related work with respect to each criterion; grayed boxes indicate unsatisfied criteria.

Domain	Ref	T	VD	CS	ER	CF	FC	ET
Pre	[5], 1996	–	H	–	?	– (Compression drop = 40%)	+	–
	[12, 13], 1998	–	H	–	?	?	+	–
	[14], 1999	–	H	–	?	?	+	–
	[15], 2002	–	H	–	– (>37.5%)	–	+	–
	[16], 2002	–	H	–	– 20%	+ (compression drop <5%)	+	?
	[17], 2004	–	V	–	V	+ (only lossless)	+	–
In	[19], 1995	+	–	?	–	+	–	–
	[11, 21], 2001	–	H	–	V	+	–	–
	[24], 2001	–	H	+	+ (<15%)	–	+	–
	[25], 2002	–	–	–	+	–	–	–
	[26], 2004	+	V	–	V	–	+	+
	[29], 2004	+	V	–	V	+	+	–
	[30], (2005)	–	H	–	–	+	+	–
	[31], 2006	–	H	+	+	–	–	–
Post	[32], 1995	–	–	–	– (>30%)	+	–	–
	[34], 1999	–	–	+	– (>30%)	+	–	–
	[35], 2000	–	+	–	14% to 50% (content dependent)	+	–	?
	[36], 2002	–	H	+	– (92%)	+	+	–
	[37], 2003	+	V	?	V	? (depends on encryption parameters)	+	?
	[39], 2004	–	V	–	V	+	+	–
	[41], 2006	–	H	–	– (>20%)	+	–	–
	[40], 2006	–	V	–	V	–	+	–
	[42], 2007	–	H	–	V	+	+	–

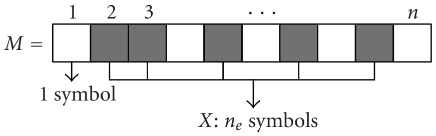

Figure 7: Selectively encrypting a message M, only gray units are encrypted.

We will evaluate the difficulty for an attacker to guess the encrypted part X in a brute force attack and try to find conditions that make brute force attack on the key space easier than optimal brute force attack on the plaintext space. We assume that the attacker knows the length and the location of the encrypted part and is able to recognize when a right guess occurs.

Perfect compression implies that all source redundancies are eliminated and that all symbols in the compressed message M are independent and identically distributed. Hence, X can be considered as a discrete random variable that takes its values in the language L^{n_e}, $X \in \{X_1, X_2 \cdots X_{|L|^{n_e}}\}$ with L being the symbols space and $|L|$ being its cardinality. The attacker would try to guess the value of X by trying all possible values in the decreasing order of their probabilities: $p_1 \geq p_2 \cdots \geq p_{|L|^{n_e}}$, the guesswork is given by

$$W(X) = \sum_{i=1}^{|L|^{ne}} i \cdot p_i. \tag{3}$$

Note that for perfect compression, all symbols are equally probable: $p_i = 1/|L|^{n_e}$, this gives a guesswork:

$$W(X) = \frac{1}{|L|^{n_e}} \sum_{i=1}^{|L|^{n_e}} i = \frac{|L|^{n_e}+1}{2}. \qquad (4)$$

Now, if we consider the guesswork on the key space (of k bits), we would have

$$W(K) = \sum_{i=1}^{2^k} \frac{i}{2^k} = \frac{2^k+1}{2}. \qquad (5)$$

From (4) and (5), we can conclude that brute force attack on the message space is harder than key guessing if $W(X) \geq W(K)$. In other terms,

$$|L|^{n_e} \geq 2^k. \qquad (6)$$

This yields a minimum number of bytes encrypted

$$n_{e,\min} \geq \frac{k}{\log_2(|L|)}. \qquad (7)$$

This result is fundamental especially for postcompression algorithms that perform encryption on entropy coded data. Since entropy coders can be considered, to a certain extent, as perfect compressors, it is required to encrypt at least $n_{e,\min}$ bytes. This minimum value gives the optimal encryption ratio while achieving cryptographic security. Such a result could be used to optimize encryption ratio in some proposals for JPEG2000 selective encryption, where selected packet data are encrypted [37, 39–42]. As codeblock contributions to packets (CCPs) are compressed independently and each CCP can be considered as "perfectly compressed," it is then required to encrypt only $n_{e,\min}$ bytes per CCP to achieve the same visual degradation while still guaranteeing cryptographic security. An important encryption ratio reduction could then be achieved.

4.1.3. Error tolerance

A main challenge in selective encryption algorithms is to design secure schemes that are error tolerant. Since most standard ciphers have strong avalanche effect, they provide poor error tolerance. Indeed, in networks prone to errors, a single bit error in the encrypted part will result in many erroneous bytes in the decrypted part. This is due to diffusion property of ciphers. Error tolerance and security seem to have antagonistic behaviors.

As a consequence, it is important to trade off security and error tolerance. It is then appreciated to avoid chaining modes of encryption algorithms [37, 41]. AES in CTR mode or any other cipher that encrypts data blocks independently offer a good balance between security and error tolerance.

4.2. Perspectives and future works

Although an important and rich variety of selective encryption algorithms have been proposed in the literature, we believe that many research areas remain open in this field.

(i) Can we design a selective encryption for any compression algorithm? We believe that some compression algorithms are more cooperative and could be better candidates for selective encryption. For example, compared to MPEG, JPEG2000 is a very good candidate to selective encryption; this is due to its flexibility (embedded encoding, block-based encryption, many progression orders, local region access, etc.). These properties can be very useful in designing a flexible selective encryption algorithm in order to meet a larger set of requirements and target more applications. In future works, we will focus on designing selective encryption algorithms for JPEG2000.

(ii) Can we build a rule of thumb to design a good selective encryption algorithm? The study we make here shows the bad choices to avoid when trying to design a selective encryption algorithm. For example, a selective encryption that relies only on random permutations is totally insecure since it is easily breakable by chosen-plaintext attacks. Energy concentration does not mean intelligibility concentration, and therefore, selectively encrypting low-frequency coefficients does not necessarily give a sufficient level of security or visual degradation.

(iii) Can we design a selective encryption that can be used in any kind of application? We believe that it is feasible to design a flexible selective encryption algorithm that is tunable and allows to trade off a certain number of parameters in order to target a large set of applications. The algorithm proposed in [26, 37] good examples.

REFERENCES

[1] C. E. Shannon, "Communication theory of secrecy systems," Declassified Report, 1946.

[2] T. Lookabaugh, D. C. Sicker, D. M. Keaton, W. Y. Guo, and I. Vedula, "Security analysis of selectively encrypted MPEG-2 streams," in *Multimedia Systems and Applications VI*, vol. 5241 of *Proceedings of SPIE*, pp. 10–21, Orlando, Fla, USA, September 2003.

[3] T. Lookabaugh, "Selective encryption, information theory, and compression," in *Proceedings of the 38th Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 373–376, Pacific Grove, Calif, USA, November 2004.

[4] T. Lookabaugh and D. C. Sicker, "Selective encryption for consumer applications," *IEEE Communications Magazine*, vol. 42, no. 5, pp. 124–129, 2004.

[5] L. Tang, "Methods for encrypting and decrypting MPEG video data efficiently," in *Proceedings of the 4th ACM International Multimedia Conference and Exhibition*, pp. 219–229, Boston, Mass, USA, November 1996.

[6] L. Qiao, K. Nahrstedt, and M.-C. Tam, "Is MPEG encryption by using random list instead of zigzag order secure?" in *Proceedings of the IEEE International Symposium on Consumer Electronics (ISCE '97)*, pp. 226–229, Singapore, December 1997.

[7] T. Uehara and R. Safavi-Naini, "Chosen DCT coefficients attack on MPEG encryption scheme," in *Proceedings of IEEE Pacific Rim Conference on Multimedia*, pp. 316–319, Sydney, Australia, December 2000.

[8] D. Socek, H. Kalva, S. S. Magliveras, O. Marques, D. Culibrk, and B. Furht, "New approaches to encryption and steganography for digital videos," *Multimedia Systems*, vol. 13, no. 3, pp. 191–204, 2007.

[9] J. Baumgartner, "Deciphering the CA conundrum," *Communications Engineering and Design*, March 2003.

[10] J.-L. Giachetti, V. Lenoir, A. Codet, D. Cutts, and J. Sager, "Common conditional access interface for digital video broadcasting decoders," *IEEE Transactions on Consumer Electronics*, vol. 41, no. 3, pp. 836–841, 1995.

[11] C.-P. Wu and C.-C. J. Kuo, "Fast encryption methods for audiovisual data confidentiality," in *Multimedia Systems and Applications III*, vol. 4209 of *Proceedings of SPIE*, pp. 284–295, Boston, Mass, USA, November 2001.

[12] C. Shi and B. Bhargava, "A fast MPEG video encryption algorithm," in *Proceedings of the 6th ACM International Conference on Multimedia*, pp. 81–88, Bristol, UK, September 1998.

[13] C. Shi and B. Bhargava, "An efficient MPEG video encryption algorithm," in *Proceedings of the 17th IEEE Symposium on Reliable Distributed Systems (SRDS '98)*, pp. 381–386, West Lafayette, Ind, USA, October 1998.

[14] C. Shi, S. Y. Wang, and B. Bhargava, "MPEG video encryption in real-time using secret key cryptography," in *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA '99)*, pp. 191–201, Las Vegas, Nev, USA, June-July 1999.

[15] M. Podesser, H. P. Schmidt, and A. Uhl, "Selective bitplane encryption for secure transmission of image data in mobile environments," in *Proceedings of the 5th Nordic Signal Processing Symposium (NORSIG '02)*, Tromsø, Norway, October 2002.

[16] W. Zeng and S. Lei, "Efficient frequency domain selective scrambling of digital video," *IEEE Transactions on Multimedia*, vol. 5, no. 1, pp. 118–129, 2003.

[17] D. Van de Ville, W. Philips, R. Van de Walle, and I. Lemahieu, "Image scrambling without bandwidth expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 6, pp. 892–897, 2004.

[18] S. Li, C. Li, K.-T. Lo, and G. Chen, "Cryptanalysis of an image scrambling scheme without bandwidth expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 3, pp. 338–349, 2008.

[19] J. Meyer and F. Gadegast, "Security mechanisms for multimedia data with the example MPEG-1 video," Project Description of SECMPEG, Technical University of Berlin, Germany, May 1995.

[20] L. Qiao and K. Nahrstedt, "A new algorithm for MPEG video encryption," in *Proceedings of the 1st International Conference on Imaging Science, Systems and Technology (CISST '97)*, pp. 21–29, Las Vegas, Nev, USA, July 1997.

[21] C.-P. Wu and C.-C. J. Kuo, "Efficient multimedia encryption via entropy codec design," in *Security and Watermarking of Multimedia Contents III*, vol. 4314 of *Proceedings of SPIE*, pp. 128–138, San Jose, Calif, USA, January 2001.

[22] D. W. Gillman and R. L. Rivest, "On breaking a Huffman code," *IEEE Transactions on Information Theory*, vol. 42, no. 3, pp. 972–976, 1996.

[23] J. Zhou, Z. Liang, Y. Chen, and O. C. Au, "Security analysis of multimedia encryption schemes based on multiple Huffman table," *IEEE Signal Processing Letters*, vol. 14, no. 3, pp. 201–204, 2007.

[24] J. Wen, M. Severa, W. Zeng, M. H. Luttrell, and W. Jin, "A format-compliant configurable encryption framework for access control of video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 545–557, 2002.

[25] A. Pommer and A. Uhl, "Selective encryption of wavelet-packet encoded image data: efficiency and security," *Multimedia Systems*, vol. 9, no. 3, pp. 279–287, 2003.

[26] S. Lian, J. Sun, and Z. Wang, "Perceptual cryptography on JPEG2000 compressed images or videos," in *Proceedings of the 4th International Conference on Computer and Information Technology (CIT '04)*, pp. 78–83, Wuhan, China, September 2004.

[27] M. Bertlisson, E. F. Brickell, and I. Ingemarsson, "Cryptanalysis of video encryption based on space-filling curves," in *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology (EUROCRYPT '89)*, vol. 434 of *Lecture Notes in Computer Science*, pp. 403–411, Springer, Houthalen, Belgium, April 1989.

[28] A. Massoudi, F. Lefèbvre, and M. Joye, "Cryptanalysis of a video scrambling based on space filling curves," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '07)*, pp. 1683–1686, Beijing, China, July 2007.

[29] M. Grangetto, E. Magli, and G. Olmo, "Multimedia selective encryption by means of randomized arithmetic coding," *IEEE Transactions on Multimedia*, vol. 8, no. 5, pp. 905–917, 2006.

[30] C. Bergeron and C. Lamy-Bergot, "Compliant selective encryption for H.264/AVC video streams," in *Proceedings of the 7th IEEE Workshop on Multimedia Signal Processing (MMSP '05)*, pp. 1–4, Shanghai, China, October 2005.

[31] D. Engel and A. Uhl, "Lightweight JPEG2000 encryption with anisotropic wavelet packets," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '06)*, pp. 2177–2180, Toronto, Canada, July 2006.

[32] G. A. Spanos and T. B. Maples, "Performance study of a selective encryption scheme for the security of networked, real-time video," in *Proceedings of the 4th International Conference on Computer Communications and Networks (ICCCN '95)*, pp. 2–10, Las Vegas, Nev, USA, September 1995.

[33] I. Agi and L. Gong, "An empirical study of secure MPEG video transmissions," in *Proceedings of the Symposium on Network and Distributed System Security*, pp. 137–144, San Diego, Calif, USA, February 1996.

[34] A. M. Alattar and G. I. Al-Regib, "Evaluation of selective encryption techniques for secure transmission of MPEG-compressed bit-streams," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '99)*, vol. 4, pp. 340–343, Orlando, Fla, USA, May-June 1999.

[35] H. Cheng and X. Li, "Partial encryption of compressed images and videos," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2439–2451, 2000.

[36] M. Van Droogenbroeck and R. Benedett, "Techniques for a selective encryption of uncompressed and compressed images," in *Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS '02)*, pp. 90–97, Ghent, Belgium, September 2002.

[37] Y. Sadourny and V. Conan, "A proposal for supporting selective encryption in JPSEC," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 4, pp. 846–849, 2003.

[38] ISO/IEC, “JPSEC commission draft 2.0,” *ISO/IEC/JTC1/SC29/WG 1*, *N3397*, 2004.

[39] Y. Wu and R. H. Deng, “Compliant encryption of JPEG2000 codestreams,” in *Proceedings of the International Conference on Image Processing (ICIP ’04)*, vol. 5, pp. 3439–3442, Singapore, October 2004.

[40] T. Stütz and A. Uhl, “On format-compliant iterative encryption of JPEG2000,” in *Proceedings of the 8th IEEE International Symposium on Multimedia (ISM ’06)*, pp. 985–990, San Diego, Calif, USA, December 2006.

[41] R. Norcen and A. Uhl, “Selective encryption of the JPEG2000 bitstream,” in *Communications and Multimedia Security*, vol. 2828 of *Lecture Notes in Computer Science*, pp. 194–204, Springer, Berlin, Germany, 2003.

[42] D. Engel, T. Stütz, and A. Uhl, “Format-compliant JPEG2000 encryption with combined packet header and packet body protection,” in *Proceedings of the Multimedia and Security Workshop (MM&Sec ’07)*, pp. 87–96, Dallas, Tex, USA, September 2007.

[43] R. Lundin, S. Lindskog, A. Brunstrom, and S. Fischer-Hübner, “Measuring confidentiality of selectively encrypted messages using guesswork,” in *Proceedings of the 3rd Swedish National Computer Networking Workshop (SNCNW ’05)*, pp. 99–102, Halmstad, Sweden, November 2005.

[44] J. O. Pliam, *Ciphers and their products: group theory in private key cryptography*, Ph.D. thesis, University of Minnesota, Minneapolis, Minn, USA, 1999.

[45] D. Malone and W. G. Sullivan, “Guesswork and entropy,” *IEEE Transactions on Information Theory*, vol. 50, no. 3, pp. 525–526, 2004.