Selective Secret Sharing Scheme for Privacy of Image and Video Compressed in MPEG-Like Formats

Cyril Bergeron, Catherine Lamy-Bergot, Wassim Hamidouche, William Puech

SELECTIVE SECRET SHARING SCHEME FOR PRIVACY OF IMAGE AND VIDEO COMPRESSED IN MPEG-LIKE FORMATS

Cyril BERGERON[†], *Catherine LAMY-BERGOT*[†], *Wassim HAMIDOUCHE*[⋆] *and William PUECH*[‡]

[†] Thales SIX France, Gennevilliers, France. Emails: firstname.lastname@thalesgroup.com
[⋆] Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, Rennes, France. Emails: firstname.lastname@insa-rennes.fr
[‡]LIRMM, Univ. Montpellier, CNRS, Montpellier, France. Emails: firstname.lastname@lirmm.fr

ABSTRACT

This paper presents a reliable image and video secret storing method, aiming at ensuring the protected storage of multimedia files without risk for their owner to have their privacy violated. It relies on a particular property (standardized in MPEG-A Part.21 VIMAF) of video and image standards such as H.264/AVC, H.265/HEVC and MPEG-HEIF: a non negligible portion of the bitstream (so-called *cipherable bits*) can be altered without modifying the decodability of the stream. Such a modification results in an image or a video that is both a valid and visually encrypted media file. Not willing to be bothered by a complex key management system, we take inspiration from Shamir's secret sharing scheme, and propose a method where the secret to be shared is the original (unciphered) set of cipherable bits, that are encoded and divided in n shares, among which $k-1$ or less do not permit to reconstruct the secret, while k or more allow to recover it. Those n shares are then used to generate n ciphered fully standard compliant versions of the media. The rightful owner of the file will easily store these n versions in various Cloud storage locations, and recover them all when the fully deciphered file will be needed, while a hacker will most generally obtain only some of those files, hence not be capable to decrypt the file.

***Index Terms*—** Multimedia Security, Privacy Protection, Secret Video Sharing, Secret Image Sharing, Selective Encryption, Cloud Storage, CSE, MPEG VIMAF, AVC, HEVC, HEIF.

1. INTRODUCTION

With the generalisation of digital cameras, mobile phones and other recorders, and their use in public and private space, the issue of privacy and capability for the end-users to store in a secured manner images or videos containing private information is needed. In 2014, with the "Celebrity photo hack" has proven that even a security concerned Cloud service could be hacked, with another similar episode following in 2017, and presumably many others. Some may argue that the best option is not to upload a content one does not want to leak outside of one's own private sphere, but the ease of use for storing, sharing with other (*a priori* selected friends) and saving without risk of data loss is a huge motivation for most people to place private information in the Cloud. As a consequence, in our modern world, the main issue is no longer the available storage space but privacy and resistance to hacking, which remains difficult for end users that are not all digital natives or simply attentive. People understand the interest of the Cloud for safe keeping but the security issues (bugs, backdoors... regularly found and used by hackers) remain, leading to wonder if the best for end users is really to blindly trust storage service to keep their private life... really private. One may be interested is a solution allowing not to store everything with the same service and apply the old saying of not putting all your eggs in the same basket. This leads us back several decades ago, when the issue was to share a secret without risking the misfortune of loosing a key, but also the risk of somebody finding the complete key with only limited partial information. In 1979, Shamir [1] and Blakley [2] designed independently secret sharing methods, that protect information by splitting it into smaller pieces (or shares) that are distributed among multiple of locations. More recently, it was extended by different Secret Image Sharing (SIS) strategies [3–6], research topic in multimedia security applied to different fields such as visual authentication and identification, data sharing, secured Cloud storage. In each case, similarly to the original Shamir approach, the original image or 'secret' is transformed into n different shadow images or image shares. Each share of the secret can be then distributed and the secret image can be reconstructed if k (or more) image shares are accessible and combined (with $1 < k < n$), while less than k image shares combinations will reveal nothing about the original image. Such a system is called a (k, n)-threshold scheme (see Fig. 1).

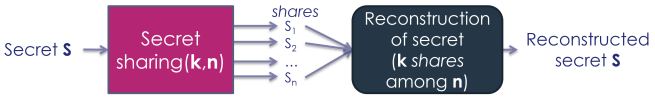


Fig. 1: (k,n)-threshold Secret Sharing scheme

Several drawbacks exit in the SIS methods proposed in the literature: they work mostly on uncompressed image files (except for [7]), the reconstruction phase can be important due to the size of shares [3], and some do not permit lossless reconstruction (though it is possible [5,6]). Furthermore, those solutions consider image only, and no application to the video field. We hence propose here to generalize the SIS approach to both image and video compressed data, taking advantage of the selective encryption method standardized in [8] that comes from a consistent work has been made last decades [9–16] in the visual cryptography field. The secret consist of the bits selected for encryption, with a lossless (k, n)-threshold scheme based on a well-known Reed-Solomon code which can be considered as a Lagrange interpolation method in $GF(2^p)$ and which is already presented as a method to satisfy the Shamir's scheme in [17]. This Secret Sharing Scheme is applicable to compressed video such as H.264/AVC [18] or H.265/HEVC [19] and compressed image format such as HEIF [20] (based on Intra mode of AVC and HEVC codecs), and can be used both to cipher either the complete stream or only selections of it (*e.g.* faces of people).

This paper is organized as follows: Section 2 introduces video encryption and summarizes the principle of Content Sensitive Encryption, followed by Section 3 which details the proposed approach, and its possible application in a secured Cloud storage. Then Section 4 presents experimental results highlighting the privacy offered by the encryption and the losslessness of the reconstruction. Finally, Section 5 draws some conclusions.

2. PRIVACY FOR VIDEO: ENCRYPTION TECHNIQUES

2.1. Video Encryption

The increased usage of image or video compression in various services (medical imaging, social networking, media sharing platforms...) has led to a high security-related issue for copyright protection, confidentiality or user's privacy. In the state of the art we find several video encryption methods that can be grouped in three major categories: Visual Encryption methods working at pixel level before compression, Crypto-Compression methods that operate while the compression is applied and take the bitstream structure into account and Naive Encryption methods that manipulate the compressed bitstream as generic data, depending on the stage of the video transmission they are applied, respectively in the Spatial-Domain, Video Coding Layer or the system level, as illustrated in Fig. 2.

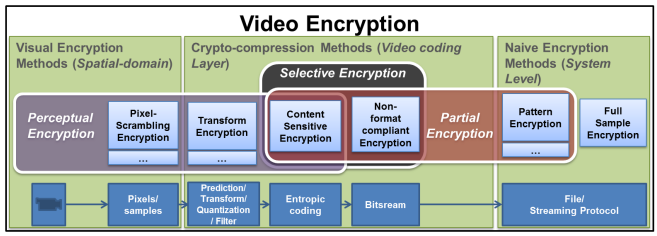


Fig. 2: Video Encryption Techniques.

Working at pixel level before compression without a specific method generally generates a compression process degradation, as the ciphering (as any other scrambling) reduces of the coherence in the image or the video stream, hence creating larger bitstreams and more complex operations of compression and file management. On the other hand, treating the compressed video bitstream as an opaque data without taking into account the compressed video structure like the *Full sample encryption* standardized in MPEG Common Encryption (CENC) [21] does present the major drawback to generate a non-negligible computational complexity as the whole video data must be in general deciphered to allow video decoding.

As a consequence, two approaches have been proposed in the literature, that rely on the knowledge of either the video structure or the stream structure to operate and ensure that said structure is maintained: the Partial and Perceptual Encryption methods. Partial encryption protects a portion (in bytes or bits) of the original bitstream while other portions are left as *plaintext* (*i.e.* unchanged). By limiting the number of bits ciphered, it reduces the complexity due to the inherent XOR operations and the usage of ciphering algorithm itself and the compression degradation due to limiting the scrambling. Perceptual Encryption operates with taking into account the stream structure, while keeping the bitstream decodable, whether to degrade some regions (*e.g. region encryption*) or the totality of the content.

2.2. MPEG Content Sensitive encryption (CSE) approach

Work on Partial or Perceptual encryption has emerged as an effective cryptographic security methods for different video and image codecs [10–15], and the MPEG committee has worked to defined a selective encryption process efficient from a cryptographic point of view but also maintaining a format compliant bitstream (perceptual encryption methods do not necessarily maintain that capability [22]) and a good compression rate [23]. As a matter of fact, degrading too much the compression efficiency would be problematic and could even be a bias for statistical attacks based on occurrence probability of compressed symbol. Some authors provided solutions for format-compliant selective encryption with bitrate constraint by modifying bits inside the entropic coding process, for instance by providing a scheme to provide selective encryption for AVC bitstreams coded with Context-Adaptive Variable-Length Coding (CAVLC) entropy coding mode by selecting some bits in different codewords in [11], or for Context Adaptive Binary Arithmetic Coding (CABAC) coding for AVC [12] and for HEVC [13]. As shown in Fig. 3, these solutions ensure bitstream format compliancy without compression efficiency loss and can be performed during the compression process (or by transcoding as proposed in [24]) which ensures a minimum latency and a negligible computing cost. Furthermore, slices (tiles in HEVC) can be used to split the video frame into areas with an integer number of blocks where Intra prediction and the entropy coding dependencies are reset at the slice/tile boundaries. Thus, as proposed in [25], it is possible to perform a selective region encryption only on specific slices and tiles. The application of ciphering on a Region of Interest (RoI) only is also illustrated in Fig. 3, where only the face of a colleague (contained in one specific tile of HEIF) is protected .

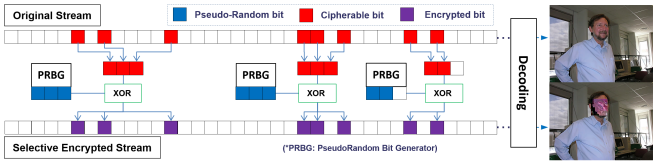


Fig. 3: Constant bitrate Selective Encryption principle

Based on these works, the *Content Sensitive Encryption* defined by MPEG standardises which part of the syntax can be 'selected for encryption' (the so called cipherable bits) for different codecs without changing the coding efficiency and without disrupting a standard decoding process [25]. In each case, the bits to be ciphered are chosen with the respect to the considered video standard to ensure full compatibility, which is achieved by selecting the bits (generally parts of code-words) for which each of the encrypted configuration modifies the decoding process contexts, which then makes it difficult for cryptanalysis attacks to find a weakness to break the cipher, as there is no redundancy to rely upon to break the code. Exhaustive definition can be found in the last edition of MPEG VIMAF [8] standard for video formats AVC, HEVC and image format HEIF, and a simple illustration is given below with example of the value of Motion Vector Difference (MVD) used in AVC with CAVLC entropic coding in Table 1. Noticing that the change of the MVD has no influence on the rest of the decoding (except visually), the suffix bits of the Exp-Golomb code are considered as cipherable bits (*i.e.* the bits following the first '1'), as illustrated by the highlighting (**bold** font).

Table 1: VLC codewords: Motion Vector Difference.

Index	Codeword	MVD
0	1	0
1	010	1
2	011	-1
3	00100	2
4	00101	-2
5	00110	3
...	...	...

In CABAC (used in HEVC and in some AVC profiles), due to the arithmetic encoding step and the Context Modeling, bits in compressed streams are very interdependent. However, only by-passed coding does not update probability models during the arithmetic coding, so that them being ciphered has not impact on the CABAC engine. Thereby, for example, signs of MVD are cipherable bits since they are binarized in Fixed-Length code and by-passed.

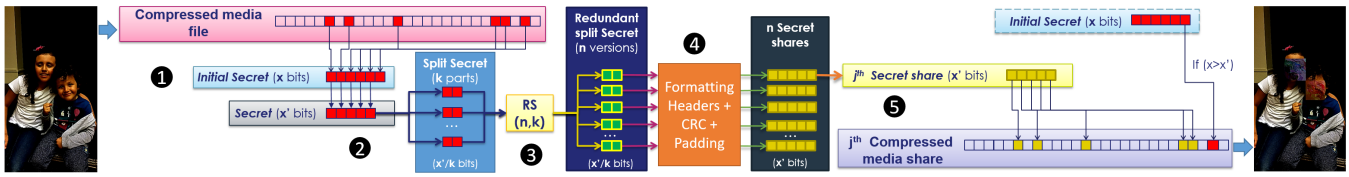


Fig. 4: Overall creation of shares scheme

3. PROPOSED METHOD FOR SELECTIVE SECRET SHARING FOR IMAGE AND VIDEO COMPRESSED FILES

The proposed method for ensuring the privacy of image or video streams stored in the Cloud is illustrated in Fig. 4, and consists of 5 different steps, that are detailed in the subsections hereafter. This can be done for complete media ciphering or partial ciphering when one uses Region selective encryption. To ensure the privacy of multimedia files stored in various public Cloud storage providers, different versions of the original file, each ciphered, are produced for storage in different Cloud services (under assumption that no transcoding is done in said Clouds), without totally relying on the security of various storage services as illustrated with 3 shares in Fig. 5. The resulting region selective encrypted pictures illustrate how the user could then browse them files without decoding when searching for particular content (*e.g.* a picture with specific attributes and given background context).

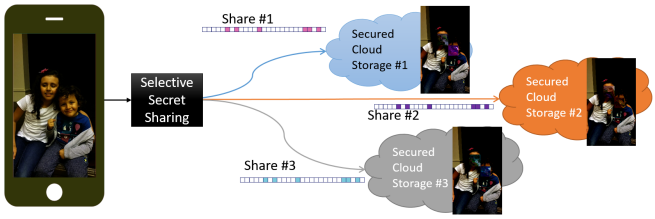


Fig. 5: Photos (and videos) back-up to Secured Cloud Storage.

Even if one of the different Cloud services (which all guarantees a given security level) is attacked by malicious persons, the media files can not be deciphered, which permits to maintain privacy for the user, while the user can access to sufficient shares (k or more) and consequently can decipher the original compressed file.

3.1. Selecting a secret for image or video compressed files

As explained in Section 2, the MPEG Content Sensitive Encryption scheme defines a set of cipherable bits, which permits to keep a compliant bitstream while ensuring that said ciphered stream is not visually meaningful for an end-user without the original cipherable bits information. Since the CSE scheme is standardized [25] and fully systematic, it is easy to find and extract those cipherable bits for each frame, slice, or tile, and the same is true of the deciphering operation. The number of bits, (noted x bits in Fig. 4-1), corresponding to the *initial secret* obviously depends on the original media and the compression parameters, but on average, for the numerous tests carried with different codecs and obtained bitrates by these authors [11–13], it was found to be roughly $15-20\%$ of the compressed media size. It should also be noted that the secret extraction can easily be parallelized by dealing with different sets of slices, tiles or pictures in parallel if needed.

3.2. Reducing the initial secret to a usable secret (divisible by k)

As with other (k,n)-threshold scheme, we want to divide the secret sequence into n shares, from which any k different shares permit to reconstitute the original secret. To ensure this, we use Reed-Solomon codes, standardly operating on Galois Field $GF(2^p)$, with $p = \lfloor Log_2(n) \rfloor + 1$ to guarantee that $k < n < 2^p$. The $RS(n,k)$ code takes as input p bits for each of the k data symbols to produce n coded symbols. Since the secret sequence length (x bits) is variable and unlikely not match systematically the Reed-Solomon symbols size, we need to systematize the reduction of the *secret* sequence to x' bits, with $x' < x$ defined as $x' = k.p\lfloor x/(k.p) \rfloor$. As represented in Fig. 4-2, the x' bits constituting the *secret*, give k data sequences of x'/k bits (*e.g.* $x'/(k*p)$ symbols in $GF(2^p)$).

3.3. Generating n sets of bits

The reason for using Reed-Solomon code, that would be true with any other Maximum-Distance Separable code (MDS code) or any perfect erasure code, is that it can correct up to $n-k$ erasures, meaning that any k encoded symbols among the total of n encoded ones allow to reconstruct perfectly the original k data symbols. This family of codes has for generator matrix a $(k \times n)$ Vandermonde matrix, which encodes a message M $(m_0, \ldots, m_{k-1})$ made up of k element symbols of Galois field $GF(2^p)$ into n symbols of $GF(2^p)$, leading us to obtain n sequences of x'/k bits, as illustrated in Fig. 4-3 on the k sequences of x'/k bits vertically ordered. Reed-Solomon encoding corresponds to the evaluation of a polynomial $f_m(a) = \sum_{i=0}^{k-1} m_i.a^i$ and from the Lagrange Interpolation formula, we know that the polynomial $f_m(a)$ can be obtained from any k of the n points (since the polynomial $f_m(a)$ is of degree $k-1$) if and only if the n points are distinct (*i.e.* $a_1, ...,$ and a_n are distinct). An advantage of Reed-Solomon codes is that they have decoders more time-efficient than Lagrange interpolation, which is useful when wanting to decode the shares.

3.4. Create formalized shares

As already mentioned, this approach does not aim at reducing Cloud storage space but at ensuring privacy for the user, and as such we accept the cost of storing more than the original compressed file. In our method, we consequently generate n compressed files, that each differ by the values of their secret share. Having currently n sequences of x'/k bits (as [6]), we need to expand these sequences to obtain for each a new sequence of x' bits to substitute the cipherable bits in the original compressed file as shown in Fig. 4-4. For that, we propose to consider $GF(256)$ and to first add a header containing the value of a_j used for the j^{th} share and the parameter k coded on $p = 8$ bits each. Then we concatenate the generated j^{th} share on x'/k bits, with also adding a cyclic-redundancy check (CRC) computed on previous bits to check for any error and prevent trying to decode with an erroneously transmitted share. Finally, we pad with pseudo-random generated bits to obtain the desired expanded size (*i.e.* from x'/k bits to x' bits) as shown in Fig. 6.

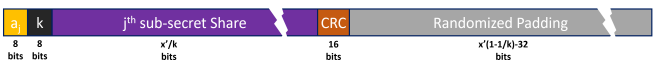


Fig. 6: Formalized shares.

It should be noted that in lieu of some padding bits, one could also add other information such as a media identifier (UUID), the value of p and the primitive polynomial of the Galois Field.

3.5. Create Compressed media shares

Remains now to generate the n different ciphered streams by replacing the initial cipherable bits by the x' different shares. As shown in Fig. 4-5 we represent the creation of this j^{th} compressed media share without forgetting to keep the last original bits if $x' < x$. In Fig. 4, we illustrate application of Region-selective encryption where only tiles (containing people's faces) are ciphered by CSE [25], but naturally the method is the same when the whole image or video stream are ciphered. We want to emphasize that this method permits us not to indicate which parts of the content are encrypted (even with Region selective Encryption), as the decoder will know it when comparing whether the cipherable bits are the same or not from two different media shares. Note also that by using CSE principle, the size of media shares is exactly the same as the original compressed file.

4. IMPLEMENTATION RESULTS

Our implementation is based on the reference software of MPEG-A VIMAF [26], which allows to automatically select the cipherable bits (approximately 15% to 25% of the compressed bitstream depending on the considered bitrate). First application was done with $n = 7$ files decodable by any compliant HEVC decoder, tested on seven complete reference video sequences (compressed with $QP = 22$) with operating on $GF(256)$. For each, objective metrics were used to measure the ciphered videos quality: Peak Signal to Noise Ratio (PSNR), Structural Similarity (SSIM) with is strongly linked to the degree of perceptual encryption with the video quality levels [27] and Edge differential ratio (EDR) [28] which measures deviation in the location of edge formation contributing pixels in the original image and its encrypted video frame (a value close to 1 corresponding to a high edge hiding capability). The PSNR, SSIM and EDR results for original (with lossless reconstruction) and Selective Secret Sharing schemes are provided in Table 2 for the seven reference video sequences and show the good performance.

Table 2: PSNR, SSIM & EDR values of original/reconstructed HEVC video and the average of different selective video shares.

Test Sequences (QP=22)	original		ciphered versions		
	PSNR	SSIM	PSNR	SSIM	EDR
PeopleOnStreet	42.8	0.96	11.2	0.17	0.92
Kimono	43.2	0.97	9.9	0.22	0.85
ParkScene	42.3	0.97	10.7	0.2	0.88
Cactus	41.5	0.96	8.4	0.23	0.83
BQTerrace	40.8	0.96	7.8	0.21	0.9
BasketballDrive	41.5	0.98	10.1	0.23	0.87
Vidyo1	45.2	0.99	11.3	0.17	0.85

Visual results are given in Fig. 7 for the 1^{st} frame of *Cactus* video sequence reconstructed for each of the different 7 generated ciphered streams, as well as the reconstruction of the compressed bitstream when recovering $k = 3$ or more streams. It illustrates the results provided in Table 2 since the structural information of encrypted frame are completely hidden and become useless for the attacker.
Furthermore, an example of visual result obtained on a HEIF image format (compressed in HEVC Intra mode) when restricting the privacy to a part of the media content (applied only on the 2 specific tiles containing faces) is presented with $n = 4$ shares in Fig. 8.

5. CONCLUSION

In this paper, we have described a reliable image and video secret sharing method relying on MPEG Content Sensitive Encryption standardised approach and based on Shamir's secret sharing principle, to allow for private compressed media files storage in the Cloud. This method permits to reinforce the Cloud storage providers security without having to manage secret keys. It works for full media ciphering but also for region only ciphering (for example by ciphering only human faces) by applying Region-selective encryption scheme. Compared to other solutions in the literature which mostly consider uncompressed images, this method allows to protect images and videos compressed by modern and widely used codecs, without changing the compression efficiency through the whole process. Furthermore, the objective analysis results presented demonstrate the effectiveness of the proposed Selective Secret Sharing methods, confirming the interest of such an encryption scheme in media Cloud storage applications.

Further works will consider how this method can be used with reducing the data to be downloaded to decipher the protected media while in mobility, typically by recovering only one complete share and part of the others (the cipherable bits, *i.e.* approximately 20% of total bitrate) even with Region Encryption coding.

6. ACKNOWLEDGEMENT

The authors would like to express their respectful thanks to their former colleague Pierre-André Laurent for years of patient support.

7. REFERENCES

[1] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, p. 612–613, Nov. 1979.

[2] G. R. Blakley, "Safeguarding cryptographic keys," in *1979 International Workshop on Managing Requirements Knowledge (MARK)*, 1979, pp. 313–318.

[3] C.-N. Yang, "New visual secret sharing schemes using probabilistic method," *Pattern Recognition Letters*, vol. 25, no. 4, pp. 481–494, 2004.

[4] F. Liu, C. K. Wu, and X. J. Lin, "Colour visual cryptography schemes," *IET Information Security*, vol. 2, no. 4, pp. 151–165, 2008.

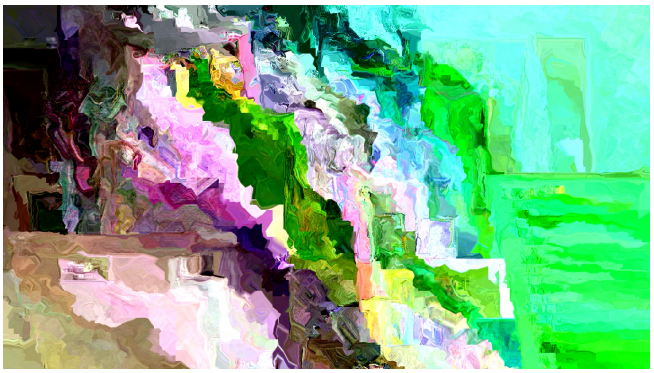
[5] P. Li, P.-J. Ma, X.-H. Su, and C.-N. Yang, "Improvements of a two-in-one image secret sharing scheme based on gray mixing model," *Journal of Visual Communication and Image Representation*, vol. 23, no. 3, pp. 441–453, 2012.

[6] C.-C. Thien and J.-C. Lin, "Secret image sharing," *Computers & Graphics*, vol. 26, no. 5, pp. 765–770, 2002.

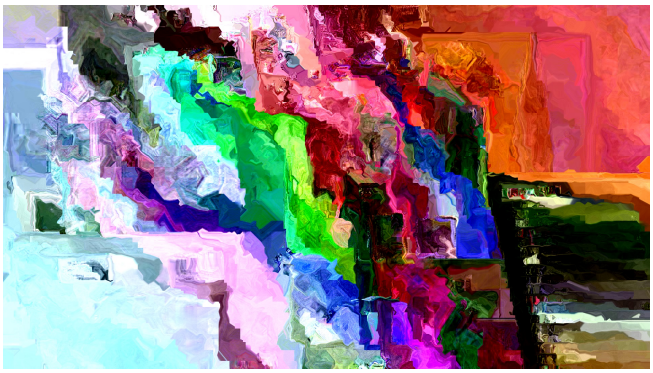
[7] F. Yriarte, P. Puteaux, and W. Puech, "A joint secret image sharing and JPEG compression scheme," in *IEEE International Conference on Image Processing (ICIP)*, October 2022.

[8] ISO/IEC 23000-21, "Multimedia application format (MPEG-A)—part 21: Visual identity management application format (VIMAF)," *ISO/IEC JTC 1/SC 29/WG11 (MPEG)*, 2019.

[9] C. Shi and B. Bhargava, "An efficient MPEG video encryption algorithm," in *Proceedings Seventeenth IEEE Symposium on Reliable Distributed Systems*, 1998, pp. 381–386.



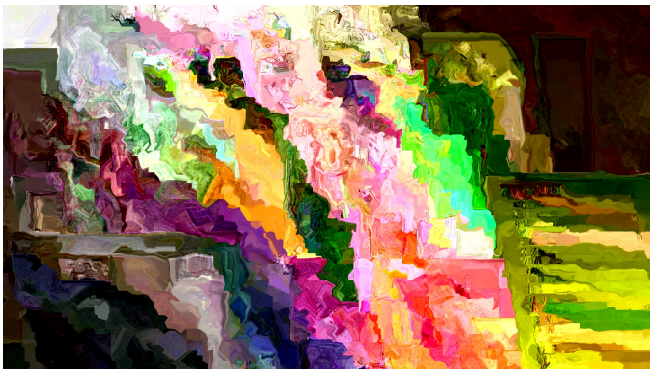
(a) 1^{st} share (PSNR=8.97 dB)



(b) 2^{nd} share (PSNR=7.98 dB)

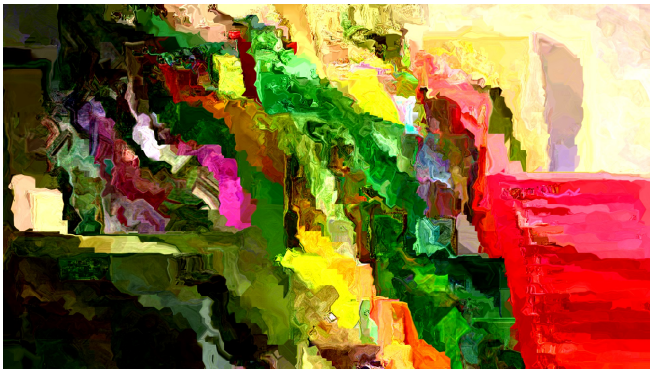
(c) 3^{rd} share (PSNR=7.3 dB)

(d) 4^{th} share (PSNR=9.12 dB)



(e) 5^{th} share (PSNR=8.23 dB)

(f) 6^{th} share (PSNR=7.52 dB)



(g) 7^{th} share (PSNR=8.14 dB)

(h) reconstructed (PSNR=44.5 dB)

Fig. 7: Visual quality obtained for different ciphered versions of Cactus Sequence (a),(b),(c),(d),(e),(f), and (g) and the reconstructed video obtained with any 3 shares (h).

1^{st} share 2^{nd} share 3^{rd} share 4^{th} share

Fig. 8: Visual quality obtained for different shares with Region encryption scheme applied on 2 tiles to perform face scrambling on a picture

[10] F. Dufaux and T. Ebrahimi, "Scrambling for Privacy Protection in Video Surveillance Systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 8, pp. 1168–1174, 2008.

[11] C. Bergeron and C. Lamy-Bergot, "Compliant selective encryption for H.264/AVC video streams," in *IEEE 7th Workshop on MMSP*, Shangai, China, Nov. 2005.

[12] Z. Shahid, M. Chaumont, and W. Puech, "Fast protection of H.264/AVC by selective encryption of CAVLC and CABAC for I&P frames," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 5, pp. 565–576, May 2011.

[13] W. Hamidouche, M. Farajallah, M. Raulet, O. Déforges, and S. El Assad, "Selective Video Encryption using Chaotic System in the SHVC Extension," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[14] T. Stutz and A. Uhl, "A survey of H.264 AVC/SVC encryption," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 3, pp. 325–339, March 2012.

[15] B. Boyadjis, C. Bergeron, B. Pesquet-Popescu, and F. Dufaux, "Extended Selective Encryption of H. 264/AVC (CABAC) and HEVC Encoded Video Streams," *IEEE Transactions on circuits and systems for video technology*, vol. 27, no. 4, pp. 892–906, 2017.

[16] M. Farajallah, W. Hamidouche, O. Déforges, and S. El Assad, "ROI Encryption for the HEVC Coded Video Contents," in *IEEE International Conference on Image Processing (ICIP)*, 2015.

[17] R. J. McEliece and D. V. Sarwate, "On sharing secrets and Reed-Solomon codes," *Commun. ACM*, vol. 24, no. 9, p. 583–584, Sep. 1981.

[18] ITU-T Rec. H.264 and ISO/IEC 14496-10 MPEG-4 part.10, "Coding of audio-visual objects — Part 10: Advanced Video Coding (AVC)," 2003.

[19] ITU-T Rec. H.265 and ISO/IEC 23008-2, "High efficiency coding and media delivery in heterogeneous environments — Part 2: High Efficiency Video Coding (HEVC)," 2013.

[20] ISO/IEC 23008-12 , "MPEG systems technologies — part 12: High Efficiency Image File Format (HEIF)," *ISO/IEC JTC 1/SC 29/WG11 (MPEG)*, 2017.

[21] ISO/IEC 23001-7, "Common encryption in iso base media file format files," *ISO/IEC JTC 1/SC 29/WG11 (MPEG)*, 2016.

[22] S. Lian, Z. Liu, Z. Ren, and H. Wang, "Secure advanced video coding based on selective encryption algorithms," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 2, pp. 621–629, May 2006.

[23] G. Van Wallendael, A. Boho, J. De Cock, A. Munteanu, and R. Van de Walle, "Encryption for high efficiency video coding with video adaptation capabilities," *IEEE Transactions on Consumer Electronics*, vol. 59, no. 3, pp. 634–642, 2013.

[24] B. Boyadjis, M.-E. Perrin, C. Bergeron, and S. Lecomte, "A Real-Time Ciphering Transcoder for H. 264 and HEVC Streams," in *IEEE International Conference on Image Processing (ICIP)*, 2014.

[25] C. Bergeron, N. Sidaty, W. Hamidouche, B. Boyadjis, J. Le Feuvre, and Y. Lim, "Real-time selective encryption solution based on ROI for MPEG-A visual identity management AF," in *22nd International Conference on Digital Signal Processing (DSP)*, 2017, pp. 1–5.

[26] C. Bergeron, W. Hamidouche, J. L. Feuvre, and B. Boyadjis, "Content sensitive encryption: Break the CSE challenge," *http://openhevc.insa-rennes.fr/press-release/avc-and-hevc-selective-video-encryption-decryption-challenge/*, 2017.

[27] S. A. Fezza, W. Hamidouche, R. A. Kamraoui, and O. Déforges, "Visual security assessment of selective video encryption," in *2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*, 2019, pp. 1–3.

[28] X. Zhang and X. Wang, "Chaos-based partial encryption of SPIHT coded color images," *Signal Process.*, vol. 93, no. 9, p. 2422–2431, sep 2013.