EURASIP Journal on Information Security
a SpringerOpen Journal

RESEARCH Open Access

A detailed evaluation of format-compliant encryption methods for JPEG XR-compressed images

Stefan Jenisch and Andreas Uhl*

Abstract

JPEG XR is the most recent still image coding standard, and custom security features for this format are required for fast adoption of the standard. Format-compliant encryption schemes are important for many application scenarios but need to be highly customised to a specific recent format like JPEG XR. This paper proposes, discusses, and evaluates a set of format-compliant encryption methods for the JPEG XR standard: *coefficient scan order permutation, sign bit encryption, transform-based encryption, random level shift encryption, index-based VLC encryption,* and *encrypting entire frequency bands* are considered. All algorithms are thoroughly evaluated by discussing possible compression impact, by assessing visual security and cryptographic security, and by discussing applicability in real-world scenarios. Most techniques are found to be insecure and, in a cryptographic sense, have a limited range of applicability and cannot be applied to JPEG XR bitstreams in an efficient manner. Encrypting entire frequency bands is identified to be a good solution in case a weaker form of format compliance can be accepted.

1 Introduction

Encryption and compression algorithms share one big commonality. Both produce high entropy output. Compression aims for reducing content size by removal of redundancies leading naturally to high entropy output. Encryption tries to hide content by transforming data into high-entropy data streams.

Usually, compression and encryption algorithms are combined by first compressing the content and encrypting it afterwards. Clearly, this is because compressing encrypted content is hardly possible due to its high entropy nature.

The most secure approach to encrypt any media format, also referred to as the 'conventional' encryption approach, is to encrypt the entire compressed bitstream with a secure cipher, e.g. AES, in a secure mode, e.g. cipher block chaining (CBC).

However, there are well-founded reasons not to stick to this approach but to apply specifically designed encryption routines:

1. The implementation of advanced application scenarios, such as secure adaptation, transparent/perceptual encryption, and privacy preserving encryption in video surveillance
2. The preservation of certain properties and functionalities of the bitstream, such as format compliance, scalability, streaming/packetisation, fast forward, extraction of subsequences, transcodability, watermarking, and error resilience
3. The reduction of computational complexity (especially in the context of mobile computing)

In many of those specifically designed encryption routines, techniques like lightweight/soft/partial/selective encryption are employed, which achieve their respective advantages with a loss in security/secrecy as compared to conventional encryption. According to the properties of such schemes, the following application scenarios of media encryption schemes may be distinguished:

- *Cryptographic encryption*: no information about the plaintext (image and compressed file) shall be deducible from the ciphertext.

*Correspondence: uhl@cosy.sbg.ac.at
Department of Computer Sciences, University of Salzburg, Jakob Haringer Str. 2, Salzburg 5020, Austria

- *Content security/confidentiality*: information of the plaintext may leak, but the visual image content must not be intelligible/discernible.
- *Sufficient encryption*: The content must not be consumable due to high distortion (DRM systems); it is sufficiently protected to prevent an enjoyable viewing experience.
- *Transparent/perceptual encryption*: Reducing content quality but maintaining a certain quality level so that it can be used as a low-quality preview and still attracts potential customers.

Format compliance is a key property for many of the functionalities and properties listed so far. Therefore, the research in the field of format-compliant encryption is of significant interest and has been thoroughly conducted in the past. In the case of discrete cosine transform (DCT)-related block-transform-based compression standards, several format-compliant encryption strategies have been developed using the JPEG or MPEG algorithms as a case study, e.g. *coefficient scan order permutation* [1], *sign bit encryption* [2], and *index-based variable-length coding (VLC) encryption* [3]. A method based on alternative block transforms, here referred to as *transform-based encryption*, was presented in [4] and [5] for the H.264/AVC standard.

JPEG XR [6] is the most recent in a series of ISO/ITU still image coding standards aiming to keep the simple structure of baseline JPEG but offering some of the advanced JPEG2000 features as well. JPEG XR has been originally derived from the Microsoft HD Photo format and is primarily designed for the efficient compression of still tone images. One of the main design goals was to provide support for up to 16 b per colour channel allowing high dynamic range imaging and true lossless compression.

The standard also supports tiling, which allows the image to be partitioned into independently processable segments.

The frequency transformation is DCT based and uses transform blocks (similar to JPEG). The transform blocks are called macroblocks and have a size of 16×16 pixels. They are partitioned into atomic transform blocks of a size of 4×4 pixel. The transform process consists of two stages. In the first stage, these atomic blocks are transformed into the frequency domain. Second, the DC portions of each block are grouped together and again are subject to the frequency transform. The result of this process is one DC, 15 lowpass (LP), and 240 highpass (HP) coefficients for each macroblock. The grouping of the coefficients form three frequency bands: the DC, LP, and HP, respectively.

The HP coefficients are further processed by splitting the coefficients at bit level into a variable-length coded portion carrying the most significant bits and a fixed length coded portion carrying the least significant bits of the coefficients. The later are called 'FLEXBITS'.

The frequency transform used in JPEG XR is the Photo Core Transform (PCT). The transform relies completely on integer operations and allows for a lifting scheme like implementation.

To address the blocking effect, a second block transformation besides PCT was defined as 'photo overlay transform' (POT). The POT is optionally applied to the image, but its transformation grid is shifted vertically and horizontally by 2 pixels relative to the PCT grid.

The JPEG XR standard allows the organisation of the code stream in two different modes. The organisation in 'spatial mode' and the organisation in 'frequency mode'. In both modes, the image data is organised according to the tile sequence (from left to right and top to bottom) and preceded by the image header information and the index table (containing offset positions of the tiles in the code stream). The modes differ in the way the data is organised inside the tiles.

In spatial mode, the data is organised according to the macroblock sequence from left to right and top to bottom.

In the frequency mode, the data is ordered according to their frequency band, i.e. DC, LP, and HP, which are then transmitted according to their importance starting with the low-frequency portions. This allows for a low-quality preview image during transmission similar to progressive JPEG. For the subsequent discussions, we denote the number of 4×4 pixels atomic transform blocks in a given image by M, $n_t = 16M$ is the total number of coefficients found in the image, and n_i is the number of non-zero coefficients in the atomic transform block i.

While for JPEG, JPEG2000, and several video coding standards, a wide variety of encryption techniques and corresponding standardisation like JPSEC and IPMP have been proposed and assessed in detail; the respective coverage of JPEG XR is still in its infancy. Apart from [7,8], where coefficient scan order permutation, sign bit encryption, and random level shift encryption have been proposed for JPEG XR encryption, no further JPEG XR-specific encryption methodology has been considered so far. Also, with respect to standardisation of security mechanisms in JPEG XR, no efforts are conducted.

Here we discuss and evaluate a set of encryption techniques, all of which are JPEG XR format compliant. Apart from the three techniques previously published in the context of JPEG XR (as given above), we adopt/discuss transform-based encryption and index-based VLC encryption to/in the JPEG XR case and we introduce a new bitstream-oriented JPEG XR encryption technique termed *encrypting entire frequency bands*.

Where appropriate, we highlight the differences of the respective approach when applied to JPEG XR instead to earlier formats. All algorithms are thoroughly evaluated by discussing possible compression impact, by estimating the provided security level (e.g. available key space, eventual security breaches), and by assessing their respective applicability in real-world scenarios. Last but not least, format-compliant encryption methods allow for a visual security evaluation of encrypted images (i.e. the determination of visual quality and the intelligibility of visual content). For this purpose, besides providing visual examples of encrypted imagery, we use objective image quality metrics in combination with a public image database to assess the visual security of encrypted images and we rate the extent of control an encryption scheme provides to generate various levels of content protection.

The subsequent section describes the different encryption techniques considered, where potential compression impact and available key space is discussed for each technique. In Section 3, visual security evaluation is conducted (i.e. by visual examples and objective quality metrics), including a short description of the objective metrics used. Section 4 discusses real-world applicability of the algorithm while an analysis of cryptographic security is conducted in Section 5. The paper is concluded in Section 6.

2 Format-compliant encryption for JPEG XR

This section describes the considered set of format-compliant encryption techniques for JPEG XR imagery. While format-compliant encryption is usually defined by requiring all syntax definitions of the file format to be obeyed after the encryption process (as followed by subsequent techniques in Sections 2.1 to 2.4), an alternative (weaker) definition is to require the reference software to be able to decode the encrypted bitstream properly (without decoding errors, see technique in Section 2.6).

Generally speaking, encryption can be implemented at the end of the processing pipeline at bitstream level (causing parsing/decoding effort to identify bitstream parts subjected to encryption) or implemented in a compression-integrated fashion (encryption is performed at some stage of the encoding pipeline). In the latter approach, format compliance comes naturally while for bitstream encryption, format compliance needs to be assured explicitly by the applied technique.

However, the compression-integrated strategy has significant disadvantages with respect to applicability - it cannot be applied in application scenarios in a sensible manner where already compressed data is encrypted and subsequently retrieved or transmitted (i.e. off-line applications [9]), and moreover, existing compression hardware cannot be used. Thus, bitstream level encryption is desirable in most application contexts.

Bitstream level encryption requires a certain amount of parsing/decoding of the JPEG XR data in order to be able to access the entities of the format encryption is being applied to. In Figure 1, a graphical presentation of the JPEG XR parsing and decoding process is shown.

2.1 Coefficient scan order permutation

For JPEG XR-compressed images, Sohn et al. proposed a method for encryption of the LP frequency band in [7]. In this paper, we extend the method for encryption of the HP frequency band as well. The coefficient scan order permutation encryption strategy (CSOP) changes the coefficient scan order for discrete cosine-transform-based compression standards. When adapting the method to JPEG XR, which was initially proposed for encryption of MPEG-compressed frames [1], a number of problems arise.

First, the JPEG XR standard specifies an algorithm that adapts and optimises the coefficient scan order during encoding according to the image data. Clearly, the adaptation of the scan order has to be deactivated for the encryption.

Second, in case of the JPEG XR standard, only the coefficients within their own frequency band and macroblock are transposed. Inter-frequency band transposing would raise the computational requirements and parsing efforts required.

As a result, only the LP and HP coefficients are subject to this encryption method, leaving out the DC band since there exists only one DC coefficient per macroblock.

Another problem arises when all 16 coefficients of a PCT transform block, non-zero and zero ones, are subject to the permutation process. Format compliance of the code stream may not be achieved because synchronisation of the code stream is likely lost in the refinement bits or FLEXBITS portion of a coded macroblock due to a lost sign bit as it is explained in the following.

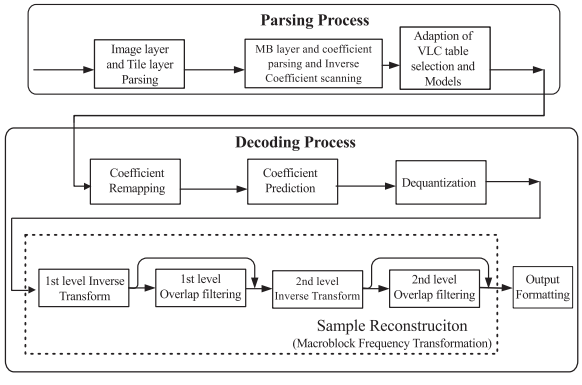

Figure 1 The JPEG XR parsing and decoding process [10].

In JPEG XR, all coefficients are divided into a VLC portion and a fixed length coded (FLC) portion. If the magnitude of a coefficient is below a particular threshold, the coefficient is completely coded in the fixed length coded portion along with its sign bit. If the magnitude of a coefficient is above the threshold, the coefficients' sign bit is coded along with the variable-length coded portion.

Also the scan order of the VLC and FLC coded part differs. While the VLC coded part uses an adaptive scan order, the FLC scan order is fixed and goes from left to right and top to bottom.

Due to the rearrangement of the permutation operation, it may happen during decoding of the encrypted code stream that a coefficient coded in VLC and FLC part merges with a coefficient solely coded in the FLC part. In that case, one of the now duplicate sign bits will be lost when decoding the encrypted code stream and thus losing synchronisation.

When synchronisation of the code stream is lost, the further decoding will most likely go wrong. In case of the JPEG XR reference software, most of the time decoding fails because the software cannot recover the end of the current transform block correctly and decoding is terminated.

One possible solution to this would be to adapt the ordering of the FLC coded part according to the modifications of the VLC coded part. Naturally, this would require additional processing of the code stream. Due to that, the adaptation of the FLC coded part has not been taken into account here. As a result, certain restrictions have to be made when utilising this form of encryption strategy.

For the LP band in frequency store mode, it is sufficient to restrict permutation only to non-zero coefficients. This is necessary since the LP refinement bits (the FLC coded part of the coefficients) immediately follow the VLC coded part.

For the HP band in frequency store mode, even non-zero and zero coefficients can be swapped. This is because the refinement bits or FLEXBITS of the band are stored as a separate chunk at the very end of the code stream.

Synchronisation still is lost during parsing the encrypted FLEXBITS portion of the code stream but this has no impact on format compliance except some lost sign bits and some wrong refinement values added to the HP coefficients.

In the spatial mode, further restrictions must be made to achieve format compliance. In case of the LP band, the coefficients with numbers 1, 2, 3, 4, 8, and 12 have to be treated with care because these coefficients are used to determine the prediction mode for the HP frequency band. Depending on the prediction mode, a different scan order is used for HP coefficients. The permutation of these LP band coefficients may lead to a premature switch of the prediction mode, resulting in a scan order that may lead to a lost sign bit. This is because in spatial mode, the FLEXBITS are not shifted to the end of the code stream but follow the VLC coded part immediately in the code stream. So either these coefficients are excluded from the permutation process in any case or only subjected to permutation if prediction mode selection is not affected. To keep encryption simple and fast (which is an important aim of these techniques), we opt for the first alternative, reducing the key space of course.

For the same reason, only non-zero HP coefficients are allowed to be subject to the permutation process in spatial mode.

Naturally, there is an impact on the filesize when using this method of encryption. To demonstrate this, experiments using the Kodak Image Database (Section IV.B) have been conducted. When compressing these images with the JPEG XR reference software in lossless mode, the average filesize using spatial store mode or frequency store mode is roughly 535 KB. When enabling encryption for the LP and HP frequency bands, using individual keys for each image, the average filesize increases around 4 KB when using spatial store mode and increases about 22 KB for frequency store mode.

The key space of the encryption method is dependent on the number of non-zero coefficients n_i in each transform block i and the number of atomic transform blocks M in the compressed image (in case different permutation keys are used for each block). M is of course dependent on the size of the image and colour bands present.

For each atomic transform block, there are 15 coefficients available for the permutation process. If all of them are subject to the permutation process, this leaves $15! = 1,307,674,368,000$ possible scan orders per transform block.

Unfortunately, this number decreases drastically when taking the above-mentioned restrictions into account; e.g. in spatial store mode, only manipulation of non-zero coefficients in the LP and HP frequency bands is allowed to maintain format compliance, but additionally the coefficient numbers 1, 2, 3, 4, 8, and 12 are excluded from the permutation process. This results in a maximum of 9! possible scan orders per transform block given the case that all coefficients are non-zero. Overall, we result in a key-space-sized $Mn_i!$ for frequency store mode and accordingly smaller for spatial store mode, using distinct keys for each atomic transform block.

2.2 Sign bits encryption

Sohn et al. presented *Sign Bits Encryption* (SBE) in [7] as an encryption method for JPEG XR. Originally, Bhargva et al. proposed the method for MPEG in [2].

Sohn et al. applied the method to a single frequency band only. Here the method is extended and applied to all frequency bands of JPEG XR. Additionally, an encryption

mode is investigated where not all coefficient signs are encrypted but only a certain percentage of all. This mode allows for a finer adjustment of the visual security and image quality degradation during encryption. Along with the reduction of encrypted bits in the code stream, also computational efforts are reduced.

Also the encryption of the sign bits is not straightforward but requires the usage of JPEG XRs frequency store mode to retain format compliance for DC and LP coefficient encryption (more details are explained below).

In JPEG XR for all non-zero coefficients, a sign bit is put into the code stream or file. In a JPEG XR stream, these sign bits show a close to uniform distribution, which implies that they have a high entropy. Encrypting these bits does not change the size of the code stream. Also SBE for non-zero coefficients can be implemented on parser level without causing a loss of synchronisation of the VLC engine. Such a loss of synchronisation would ultimately lead to a non-format-compliant code stream. Due to this properties, the sign bits are ideal candidates to be subjects of a partial encryption.

Concerning the amount of data that has to be encrypted, the portion of encrypted bits for each frequency band is different of course. When compressing the Lena image using lossless mode, there are about 3,000 sign bits for the DC band, about 41,000 sign bits for the LP band and 547,000 sign bits for the HP band in the code stream. The total size of the code stream for the Lena image is about 461 KB. This implies that if sign bit encryption is restricted to the DC band, only 0.08% of data is subject to the encryption procedure. The portion that is subject to the encryption in case of the LP sign bits in the code stream amounts to 1.11%, while the portion of HP sign bits amounts 14.8%. Naturally, these amounts increase when quantisation is turned on because of the decreasing filesize. Note that the number of sign bits only changes when non-zero coefficient turns 0 due to the quantisation.

SBE requires frequency store mode when manipulating the DC and LP band coefficients sign bits without hassle. This is due to the fact that certain LP band coefficients are used for determining the prediction mode used for the HP band coefficients. Depending on the prediction mode, one of two possible scan orders for the HP coefficients is selected. When decoding the encrypted code stream, it is possible that the wrong scan order is selected because of the encrypted sign bits in the LP band coefficients. As a result, the synchronisation is lost when using spatial store mode in the same way as used for the CSOP encryption method. Spatial store mode can be used for SBE when restrictions are made (similar restrictions as made for the CSOP encryption), where scan order selection is not affected. But since this significantly increases encryption complexity because of the required analysis stage, we restrict the application of SBE to frequency store mode.

The key space of the method depends on the number of non-zero coefficients in the code stream (Mn_i) and is upper bounded by 2^{Mn_i} in case a one-time pad is used; in practice, the key space is determined by the key space k of the technique used to encrypt the sign bits.

2.3 Random level shift encryption

Sohn et al. proposed random level shift encryption (RLSE) for JPEG XR in [8]. The suggestion was to encrypt the DC coefficients only. Here, we use and examine this method for the encryption of the other two types of coefficients (LP and HP) found in JPEG XR. Because the encryption method manipulates the coefficient magnitude, it has a significant impact on compression performance but also allows for fine-grained adjustment of the visual security level.

The coefficients are encrypted by altering their value by adding or subtracting a number. The number and the operation (subtraction or addition) is derived from the key.

The implementation of this encryption mode uses three parameters: the maximum shift value, which allows to restrict the number by which the coefficients are shifted (max shift); the percentage of coefficients that are to be randomly selected and altered; and the frequency band that should become subject of the RLSE.

A major drawback of this encryption mode is its impact on code-stream size when manipulating the HP frequency band. In Figure 2, the impact on the filesize for an increasing number of manipulated coefficients is shown, which can actually triple the filesize for HP band encryption. Also when manipulating the DC or LP frequency band, a moderate impact on code-stream size can be seen.

The two other variants displayed in Figure 2 show the filesize impact for varying the number of encrypted coefficients in all three bands concurrently, setting the maximum shift to 128 for all three bands or setting a band-specific maximum shift value. While the latter technique behaves less critically, still we observe a filesize increase up to 100% when encrypting all coefficients. The reason for this poor performance of the entropy coding stage is that no longer most LP and HP coefficients are 0 (as expected by the model). Additionally, RLSE disturbs the expected coefficient order. After PCT the expected coefficient order is decreasing in magnitude, which is no longer true after RLSE. This effect, also observed for CSOP encryption, causes the coefficient prediction stage to become inefficient.

The size of the available key space can be estimated as follows. Given S is the cardinality of the set of values in [0... max shift] a single coefficient can be altered with, the key space is of size S^{n_t}.

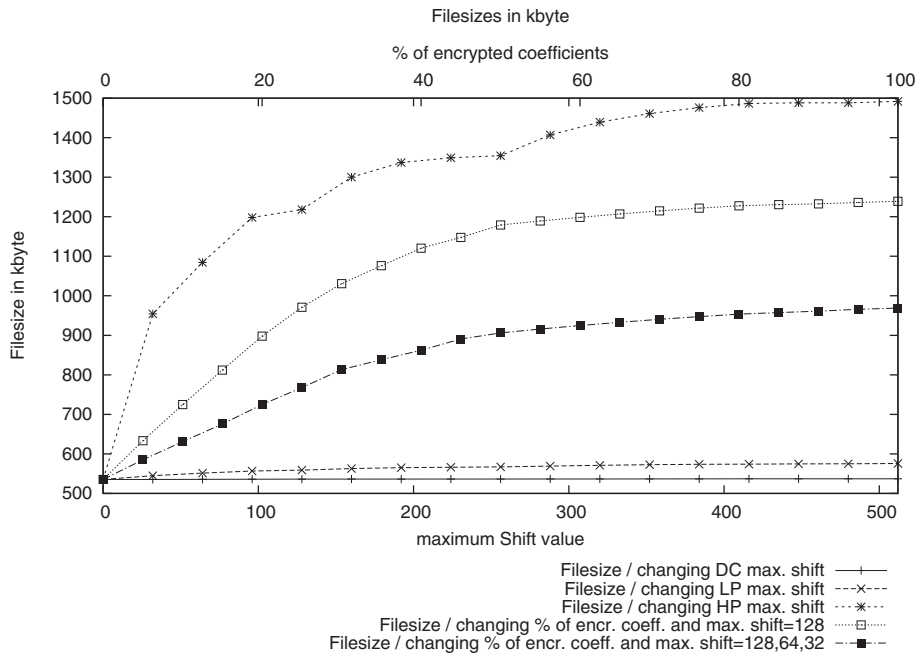

Figure 2 Averaged filesizes when encrypting the Kodak IDB images with random level shift encryption.

2.4 Transform-based encryption

Yeung et al. proposed in [4] a method for encryption of H.264/AVC encoded videos. We adapted the method to the JPEG XR standard and termed it transform-based encryption. The method uses a set of frequency transformations to translate the multimedia data into the frequency domain. During compression for each image block, one transform is selected out of the set and applied to the block. The assignment of the transforms to each block is key dependent.

We use a set of four alternative transforms that consists of DCT-II, DST-II, and two custom-made transforms. Henceforth, these latter two transforms will be called Yeung-I and Yeung-II.

This approach was adopted to JPEG XR by replacing the PCT transform. Instead of applying the PCT to a 4×4 transform block, two alternative transforms are selected from the set and applied to the block. The two-dimensional transformation is obtained by applying the first transform in the vertical and the second transform in the horizontal direction of the block.

The alternative transforms, DCT-II, DST-II, Yeung-I, and Yeung-II, all belong to the class of Fourier-related transforms. This is also true for the PCT, since it is based on the Walsh-Hadamard transform (WHT), which is also Fourier related.

The PCT is integer based and as such allows for true lossless compression when quantisation is turned off. The alternative transforms used in the experiments have been implemented using double precision floating point. These operations introduce some information loss during compression due to rounding errors. However, Yeung et al. show in [4] that the transforms can also be implemented as true lossless integer transforms.

The alternative transforms are all based on the flow graph shown in Figure 3 but use different rotation angles in the second stage.

Concerning the question of similarities between PCT and DCT, we assumed that both transforms have similar characteristics and properties with respect to signal decorrelation. We found this assumption affirmed by comparing the compression performance and image quality achieved with the PCT and DCT (the result of our experiment is shown later). Here both transforms (PCT and DCT) show almost identical curves. The plot was generated using the Kodak Image Database (IDB) and shows the averaged values of the peak signal-to-noise ratio (PSNR) image quality metric.

This is different for DST-II and Yeung-I/II transforms. These transforms do not capture the constant portion of the input signal into one single coefficient but spread the signals energy over multiple coefficients. Because of that, a phenomenon called DC leakage occurs as soon as quantisation is turned on. When combining these transforms with quantisation, a checkerboard pattern appears in the image. In Figure 4, sample images showing this phenomenon can be found. Because of the negative impact on image quality, quantisation is turned off for the encryption experiments.

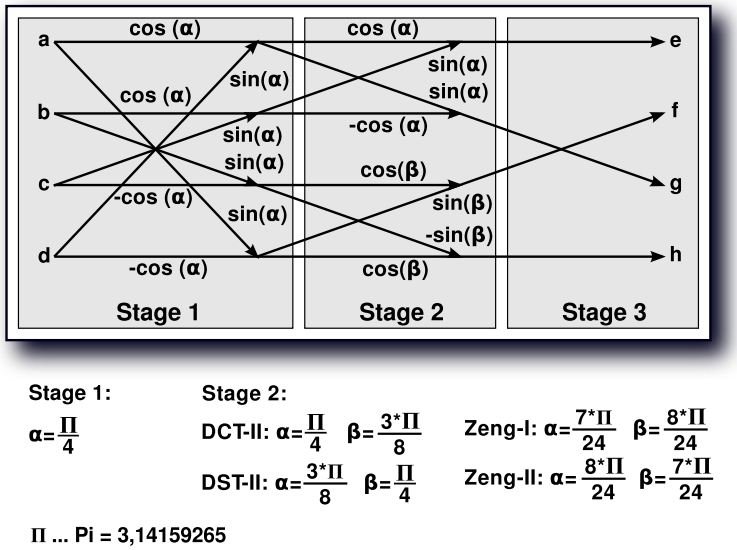

Figure 3 Flow graph for the transforms DCT-II, DST-II, Yeung-I, and Yeung-II used for transform-based encryption [4].

The compression performance achieved using only a single transform can be found in Table 1. To gather the values, again the Kodak IDB was compressed with quantisation turned off.

The transforms' impact on image quality (mostly caused by DC leakage) is plotted in Figure 5 showing PSNR values in dependence of bits per pixels.

The best performance is shown by PCT (JPEG XR default transform) and DCT-II transforms, which are almost identical. The other transforms perform not quite as good as these two. The performance ranking for these transforms is as follows: Yeung-I, than the Yeung-II, and finally the DST-II transform.

Cryptographic security (i.e. available key space) of the encryption method is dependent on image size. The number of available keys for an image is 16^M (using the set of one-dimensional transforms as discussed and a distinct key-generated transform selection for each block), where M is again the number of atomic transform blocks in the image.

2.5 Index-based VLC encryption

A method of encrypting multimedia content during the entropy coding stage (or directly at bitstream level since VLC codewords are simply exchanged) is index-based VLC encryption (IVLCE, [3]), also standardised in IPMP for MPEG-2/4. The method allows for format-compliant encryption because synchronisation of the decoding engine is not destroyed.

In this approach, the mapping between code table and symbol is dependent on and permuted according to a key. To preserve compression performance when adopting the scheme to VLC, subgroups of codewords having the same (or approximately the same) length are formed. Only the mapping between codewords and symbols of the same subgroup is affected by the key.

Due to the size and structure of the code tables in JPEG XR, the application of this approach would almost certainly have a negative impact on compression performance. Code tables in JPEG XR are small and few code words are of the same length. This requires that the index

Figure 4 DC leakage sample image using alternative transforms in JPEG XR using quantisation parameter 85. (a) Yeung-I. **(b)** Yeung-II. **(c)** DST-II.

Table 1 Average filesize and standard deviation for alternative transforms with turned off quantisation using Kodak IDB

Transform	Average filesize (B)	Standard deviation (B)
PCT	535,220.58	46,565.18
Yeung-I	596,817.58	43,109.37
Yeung-II	644,850.92	44,875.07
DST-II	690,335.83	45,423.32
DCT	546,039.38	45,785.49

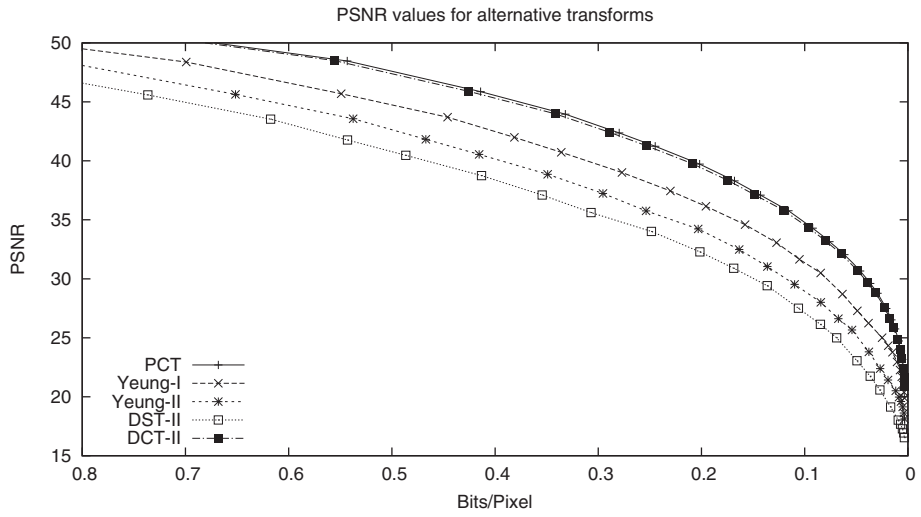

Figure 5 Image quality and filesizes for alternative transforms (Kodak IDB).

encryption uses the whole code table instead of subgroups of code words of the same length.

Additionally, the approach does break format compliance when applied to JPEG XR. This is because JPEG XR uses adaptive code table selection without indication of the code table switch in the code stream. The code tables are switched when the internal model of the coding engine indicates that another code table would be more efficient than the one currently used. The moment of code table switch is determined by looking at the decoded data the engine has seen so far.

Since the encrypted and unencrypted data decode to different symbols, the internal model of the coding engine indicates the switch of code tables for encrypted data at different positions than for unencrypted data. If code tables of encrypted data are not switched at same positions as the unencrypted data, synchronisation will be lost at this point during decoding the encrypted data. Therefore, this approach cannot be applied in the context of JPEG XR.

2.6 Encrypting entire frequency bands

Encrypting the encoded image data on a bitstream level using a block or stream cipher usually breaks the format compliance of the resulting file. This is also true when encrypting the JPEG XR code stream in such a way. Even when leaving the header information untouched and only encrypting the frequency bands (coefficient related data), it is most likely that the decoding will fail because the encrypted data will misguide the variable-length decoding engine. This usually results in an error condition of the next decoding stage of the pipeline (e.g. 18 coefficients are assigned to a transform block of the size 4×4 pixels). So the encrypted code stream is not format compliant.

We propose a method that allows to encrypt the coefficient data on the frequency band level (DC, LP, HP) with an arbitrary block/stream cipher but still remaining format compliant, termed encrypting entire frequency bands (EEFB).

When using frequency store mode, the JPEG XR standard requires that there is an index entry at the beginning of the file that contains the positions locating the frequency bands DC, LP, HP, and FLEXBITS in the code stream. These positions or offsets are used to locate the encoded coefficients during decoding.

The strategy is now to encrypt the coefficient data of a frequency band (e.g. DC, LP, or HP), generate a dummy data block used to mimic the actual coefficient data, and put this block in front of the encrypted coefficient data in the code stream. Additionally, the tile index has to be modified respectively to point to the dummy data blocks. These modifications prevent the decoding engine from decoding the encrypted data since the dummy data are decoded and the subsequent encrypted parts are ignored. In Figure 6, the principle is visualised.

One drawback of the method is the increased filesize due to the dummy data blocks. But fortunately, these dummy data blocks do not require much space if 'sparse' frequency bands are used. In these data blocks, all coefficients are set to 0.

The JPEG XR standard defines coded block patterns that makes encoding of sparse frequency bands quite efficient. These coded block patterns are placed at the start of an encoded macroblock and indicate (for all colour channels) if a coefficient is non-zero in the macroblock.

So for a dummy data block, just a sufficient number of coded-block patterns, indicating all coefficients are zero in a corresponding macroblock, has to be generated.

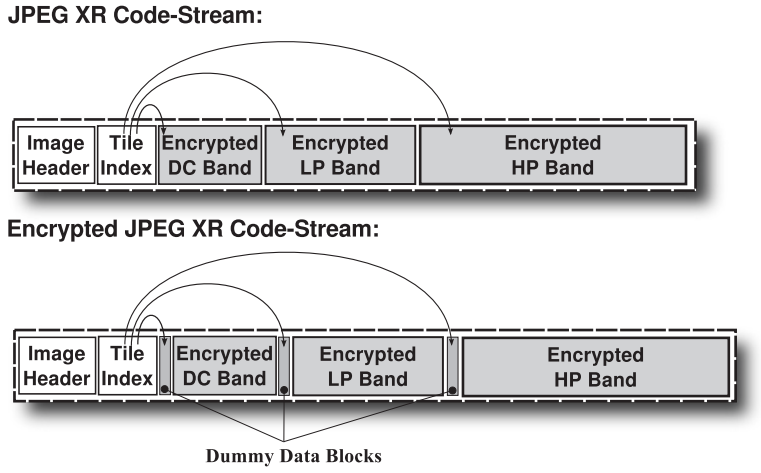

Figure 6 Encrypting JPEG XR code streams using dummy data blocks.

For the DC frequency band, a coded block pattern having 2 b is needed to indicate a macroblock where all coefficients (for all colour bands) are 0. Also the LP band requires 2 b for each macroblock (indicating that all 15 LP coefficients in all colour bands are 0) while only 1 b per macroblock is needed for the HP frequency band (indicating that all 240 HP coefficients in all colour bands are 0). The FLEXBITS can be encrypted without putting a dummy data block in front since there is no VLC involved. This segment of the code stream is only parsed by the program code responsible for the FLEXBITS; thus, possible generation of marker bits poses no problem.

Setting the dummy data block to reproduce a mean value for its corresponding frequency band effects two things. First, the filesize is minimised this way. Second, this strategy reproduces the best result of a 'replacement attack'. As the name already suggests, this attack replaces the encrypted data with NULL- or averaged data (in [9] this attack was described for bit plane encryption).

The alternative of producing a dummy data block containing disturbing image artefacts would strengthen visual security but at the cost of an increased file size. Additionally, a skilled attacker would simply run a replacement attack anyway (thus removing the artefacts causing noise in the image).

This is why encrypting all frequency bands (DC + LP + HP) produces a flat grey image with the proposed settings.

The filesize increase when applying the proposed strategy when quantisation is turned off is significantly below 1% for the Lena image; e.g. for protecting the HP band only, we obtain a filesize similar to the original size of about 461 KB, while the protection of all three bands increases the filesize by about 1 KB. Naturally, quantisation worsens the effect since the process decreases the original filesize, but the empty frequency bands stay at the same size. Still the overall filesize increase remains manageable also with significant compression.

In terms of filesize increase, the encryption of the DC band is most costly since the empty frequency band is of the biggest size when comparing it to the LP and HP frequency bands. On the other hand, the encryption of the HP band is the most expensive in terms of CPU usage since the HP band has the most coefficients.

It has to be pointed out that EEFB does not lead to a format-compliant JPEG XR file in the strict sense as opposed to the other techniques described so far since it contains encrypted parts not covered by the standard. However, a JPEG XR file protected with EEFB can be decoded by the JPEG XR reference software and any other decoding software following the required specifications. Thus, we consider this property as a 'weak' form of format compliance. The general idea of EEFB has been used for a JPEG XR region of interest encryption approach in previous work [11].

3 Evaluation of visual security

In this section, the visual security of the discussed encryption techniques is assessed, i.e. we determine if visual quality is low enough to protect content and if visual content is intelligible after encryption. As a first (subjective) stage, visual examples of encrypting the Lena test image using various parameters are given. The second (objective) stage involves three objective image quality metrics that are applied to encrypted data. Note that the data is analysed as it is given after encryption without any attacks applied. Successful attacks (see Section 5) can only improve quality, thus, the results serve as a lower bound on quality.

3.1 Subjective visual security assessment

Figure 7 shows the effect of CSOP in the LP and HP bands applied to the Lena image. Naturally, the visible effect of the HP band coefficient's permutation can only be noted when looking at the image details; e.g. in case of the Lena image, a noticeable effect can be seen along the brim. The impact on visual security when manipulating the LP frequency band is clearly more pronounced. The biggest impact on visual security is observed when encrypting both frequency bands, HP and LP.

Beside that, the more severe restrictions that have to be made to maintain format compliance in spatial mode have an impact on the visual security due to a reduced key space. When using spatial mode, visual security tends to be lower than when using frequency mode.

The extent of visual security even in the case of protecting both HP and LP bands is not high enough to provide content security, the obvious reason is that the DC coefficients are left untouched. Only the transparent and sufficient encryption scenarios can be supported.

The effects of SBE in JPEG XR can be observed on the Lena image in Figure 8. Here the signs of the DC, LP, and HP band coefficients have been encrypted using lossless mode for compression. All images have been encrypted using the same one-time pad.

The encryption of the HP sign bits is rated with the lowest visual security. Encrypting the LP sign bits is obviously stronger, followed by DC sign bit encryption. The encryption of all sign bits is rated most secure in comparison. Similar to CSOP, even the strongest form of encryption does not provide a visual security sufficient for content security. Especially, the colour clipping effect when involving DC sign bits makes this configuration hardly suited for transparent encryption.

To explore the effects of RLSE, five different settings have been singled out as case studies. First, to explore the effect of different maximum shift settings on the frequency bands, three test runs have been done where only one frequency band is manipulated with an increasing maximum shift value while the others are left untouched. Then two test runs were performed where all three frequency bands are manipulated with a static maximum shift value but an increasing amount of manipulated coefficients. The maximum shift value for one of these two test runs was set to 128 for all frequency bands and to 128, 64, and 32 for the DC, LP, and HP band in the second test run, respectively. The reduced maximum shift values for the LP and HP bands lessen the impact on code-stream size (see Figure 2).

For these encryption modes, a set of images can be found in Figures 9,10,11,12 and 13 showing the progression of the visual security at various stages when encryption parameter settings are increased.

It is interesting to note that visually, the impact of applying RLSE is strongest when the HP band is protected, followed by LP and the DC bands, respectively. This is the opposite behaviour as seen in CSOP and SBE and can be attributed to the fact that the maximum shift value used is identical for all three bands but causes highly different effects due to the different coefficient magnitudes in the three bands. However, it can be clearly seen that even when only protecting the HP band, decent visual security can be achieved (only without a targeted attack in place of course).

Figure 7 CSOP: Lena image scrambling. CSOP: Lena image scrambling using coefficient scan order permutation of HP, LP, and LP+HP frequency bands and using spatial (first line) and frequency store mode (second line).

Figure 8 Lena image scrambling using SBE. From left to right: encrypted HP band, LP band, DC band, and all bands. The colour clipping effect can be seen on the images on the right side involving encryption of the DC coefficients.

In Figures 12 and 13, it can be seen that varying the amount of manipulated coefficients is a sound approach to vary the strength of visual security. Overall, RLSE can be configured to be suited for a wide range of application contexts, ranging from content security (e.g. manipulating all bands with a high maximum shift value) to transparent encryption with small computational effort (e.g. manipulating DC band coefficients only with moderate maximum shift value, see Figure 9).

The impact on visual security when decoding a TBE-protected image using the PCT or the DCT transform can be observed in Figure 14. Here the Lena image was encrypted using a one-time-pad as a key.

The visual image quality is clearly better when decoding the encrypted image with the DCT transform than when using the PCT. This is because of the similarities among the DCT, Yeung-I, Yeung-II, and DST-II transforms and, additionally, since the DCT is part of the encryption process itself, while the PCT is not. Overall, it is obvious that visual security is not sufficient to provide content security. The effects of DC leakage and the low quality displayed in Figure 14 further hardly qualify this approach for transparent encryption, where some lower bound on image quality has to be guaranteed.

In Figure 15, the impact of EEFB on image quality can be seen when decoding the encrypted image without key.

It has to be noted that the encryption of all three bands is not visualised since the result is a uniform grey image without any structures present due to the construction of the scheme. Thus, with these settings, EEFB is suited for the content security scenario. With respect to the images shown, this method provides an exact illustration of the frequency band contents. The data of an encrypted band are simply 'ignored' and replaced by a uniform value. Therefore, subjective visual security is highest when encrypting the DC band, and lowest in case of HP encryption. Of course, encrypting both the DC and LP bands provides even higher visual security then encrypting the DC band alone. In principle, all target scenarios can be supported with this approach, however there is no way of adjusting visual security in some fine grained manner, only a small amount of settings is available to choose from.

3.2 Objective visual security assessment

The metrics used for visual security assessment are the well-known PSNR, structural similarity score (SSIM) [12], and the Local Feature Based Visual Security Metric (LFBVSM) [13].

PSNR and SSIM have been initially developed for image quality assessment in the field of image compression where SSIM was found to exhibit significantly better correlation to subjective quality ratings (MOS). However, both metrics were chosen in our context because they are commonly used for visual security assessment (like in [4]). Since the assessment of visual security obviously differs from the task of evaluating image quality in image compression, it was considered interesting to answer the question if both metrics react in the same way to the visual distortions introduced by format-compliant encryption techniques.

LFBVSM has been selected because the metric was developed specifically for the purpose of visual security assessment.

PSNR compares a reference and a target image by comparing single-pixel values while SSIM and LFBVSM compare image regions. The comparison of image regions not only allows for comparison of luminance values but also

Figure 9 DC bands encrypted with an increasing maximum shift value. From left to right: 80, 160, 280, and 2,000.

Figure 10 LP bands encrypted with an increasing maximum shift value. From left to right: 80, 160, 280, and 2000.

for incorporation of edge and contour information into the similarity evaluation.

The range of the SSIM values is the interval $[0 : 1]$ where 1 signifies identical images. The values of PSNR and LFBVSM are in the interval $[0 : \infty]$. For PSNR, the value ∞ and for the LFBVSM a value of 0 signifies identical images, respectively.

To objectively evaluate the visual security of the encryption methods, the images of the 'Kodak Lossless True Colour Image Suite' [14] or in short, Kodak IDB, were encrypted and evaluated using the objective image metrics described above. The Kodak IDB comprises 24 images with the size of 768×512 pixels.

The JPEG XR coder used for encryption was a custom-modified version of the JPEG XR reference software. The compression settings for the experiments were the default settings of the reference software varying only quantisation settings and storage mode. The photo overlay transform was restricted to the first transform stage.

The mean values and standard deviation (error intervals on top of the bars) of the objective image metrics for the CSOP encrypted Kodak Image Database images can be seen in Figure 16.

The ranking of the different settings for CSOP encryption established by the PSNR, SSIM, and LFBVSM objective image metrics is different than what would be suspected from the images in Figure 7. All three metrics indicate the lowest visual security for the encryption of the LP band in spatial mode while visually, the protection of the HP band shows the lowest security. For frequency mode, encryption of the LP band is rated as being more secure compared to the HP band, which also corresponds to subjective judgement. The tremendous visual security difference in terms of objective values between LP protection in frequency and spatial store modes can be confirmed visually on the Lena image. Also, the higher visual security of frequency mode as compared to spatial store mode in general is confirmed numerically. These first results indicate that current objective metrics in some cases exhibit low correlation to subjective visual assessment in the lower quality range.

To objectively evaluate the impact of SBE, the images of Kodak IDB are protected by encrypting all coefficients (of single bands and all bands, respectively) in the first experiment. In Figure 17, the averaged values of the objective image metrics for the encrypted images are shown along with their standard deviations.

All metrics establish the same ordering in terms of visual security for different encryption settings. The encryption of the HP sign bits is rated with the lowest visual security. Encrypting the LP sign bits is rated stronger, followed by the DC sign bit encryption. The encryption of all sign bits is rated most secure in comparison. The order can be verified subjectively when looking at the images in Figure 8.

An additional method of SBE for JPEG XR is to encrypt only a certain amount of sign bits randomly selected from the code stream. Image quality can be gradually deteriorated and adapted to various application contexts this way. In Figure 18, charts can be found visualising the averaged values of the objective image metrics for this mode of encryption using the images of the Kodak IDB. On the x-axis, the percentage of sign bits encrypted is shown, while on the y-axis, the value of the metric is found. Three different scenarios are investigated: encryption of the LP coefficients only, encryption of the LP and HP coefficients, and encryption of all frequency bands each with an increasing amount of encrypted sign bits. When

Figure 11 HP bands encrypted with an increasing maximum shift value. From left to right: 80, 160, 280, and 2,000.

Figure 12 All bands encrypted with an increasing amount of manipulated coefficients and a maximum shift value of 128. From left to right: 25%, 55%, 75%, 100%.

multiple frequency bands are encrypted, the percentage of encrypted sign bits is applied to each frequency band individually. So 10% of encrypted sign bits in the charts stands for 10% of encrypted sign bits in the DC, 10% of encrypted sign bits in the LP, and 10% of encrypted sign bits in the HP band.

All metrics show the expected behaviour and report an improved visual security when the amount of encrypted sign bits is increased. It is also evident that the joint encryption of both LP and HP bands does hardly improve the protection of the LP band encryption only. Therefore, this joint encryption should be avoided since the computational effort is significantly larger for this approach.

The evaluation of the RLSE using the Kodak IDB images is shown in Figures 19,20 and 21.

PSNR, SSIM, and LFBVSM show similar trends for the effectiveness of the encryption methods. All of them show an improved visual security when the number of encrypted coefficients or the maximum allowed shift value is increased.

Comparing the three plain encryption modes where only one frequency band is encrypted with an increasing maximum shift value, the encryption of the HP coefficients is rated as most secure by these metrics, followed by the encryption of the LP and DC band coefficients. For the two mixed modes where in all frequency bands an increasing number of coefficients is encrypted, the mode with a maximum shift value of 128 for the HP band is rated more secure than the other. Note that these results are in perfect accordance with the subjective assessment based on the encrypted Lena images shown in Figures 9,10,11,12 and 13.

The values of the objective image metrics for the images shown in Figure 14, which have been encrypted using TBE, can be found in Table 2.

The higher visual image quality of the DCT decoded images as compared to the PCT decoded ones is confirmed by the objective image metrics.

This behaviour can also be observed when looking at the averaged values when decoding the TBE protected Kodak IDB images using lossless mode. The values can be found in Table 3.

Also, the absolute value (which indicates rather low quality) of the objective metric supports the earlier observation that TBE is hardly suited for application scenarios that are different from sufficient encryption.

Finally, the averaged values and standard deviation of the objective image metrics for EEFB using the Kodak IDB can be seen in Figure 22.

The values of the image metrics suggest that encryption of the HP band offers the lowest visual security, which is in perfect accordance with visual perception (compare Figure 15). DC and LP band encryption is rated to be more secure, but the ranking is not consistent. While PSNR ranks DC encryption to be much more secure, SSIM and LFBVSM rank LP encryption as being slightly more secure. Visually, one would probably agree to the PSNR assessment due to the preservation of grey scale information in the LP protected image. The objective values obtained for the images with encrypted DC and LP bands again confirm the problems of objective visual security metrics with respect to low correlation to visual perception in some cases. Only PSNR ranks these images as being best protected, for SSIM and LFBVSM

Figure 13 All bands encrypted with an increasing amount of manipulated coefficients and maximum shift values. Maximum values are 128 (DC), 64 (LP), and 32 (HP). From left to right: 25%, 55%, 75%, 100%.

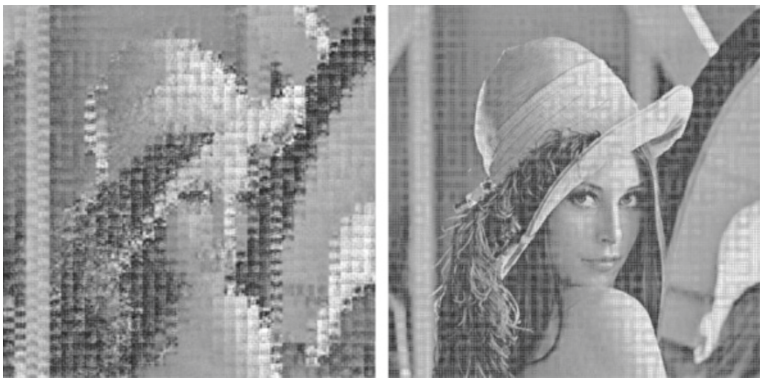

Figure 14 Transform encrypted sample images decoded using the PCT and DCT. (a) PCT decoded. **(b)** DCT decoded.

the values are in between the values of single LP and HP encryption, which does not at all correspond to the visual impression.

4 Applicability - application scenarios

4.1 Content security vs. sufficient and transparent encryption

Only RLSE as well as EEFB are able to support all three realistic target scenarios for format-compliant encryption. EEFB has a minor impact on compression performance, but RLSE can triple the filesize if not applied with care. Considering the filesizes and incorporating the subjective and objective metrics evaluation, encryption of the LP frequency band with a high maximum shift value and a variable number of encrypted coefficients to vary the degree of visual image security are considered the most effective for RLSE. However, from a security viewpoint (see below), this strategy is problematic. EEFB on the other hand does only offer a limited amount of configurations to support different application scenarios; a fine grained adjustment to specific demands is not possible using this approach.

CSOP and SBE cannot deliver content security but can be configured to support sufficient and transparent encryption scenarios. CSOP can only be configured in six different variations; thus, the number of possible degradation settings is quite limited and depending on the band protected, compression performance is impacted to some extent. SBE on the other hand, especially in its partial version, can be used with many intermediate degradation settings and does not have any impact on compression performance.

It is difficult to find any sensible application case for TBE due to the DC leakage effect as soon as quantisation is used. As a consequence of this effect, only lossless compression scenarios can be supported, and also in these cases, at most, sufficient encryption can be supported due to the rather low quality of the encrypted data. Due to the additional impact on compression performance also in the lossless case, it is questionable if this approach makes sense at all. Finally, IVLCE cannot be used in the context of JPEG XR.

4.2 Compression-integrated vs. bitstream-oriented encryption

EEFB is applied in the image layer and tile layer parsing process; thus, only little effort is required to access the entities subjected to encryption and to insert the dummy data. Thus, EEFB is well suited to be applied to bitstreams. SBE is applied after adaption of VLC table selection and models, so still before actual decoding, which makes an application of this approach to bitstreams feasible. CSOP

Figure 15 Impact on image quality when using dummy data blocks and decoding encrypted images without key. From left to right: encrypted DC, LP, HP, and DC+LP band.

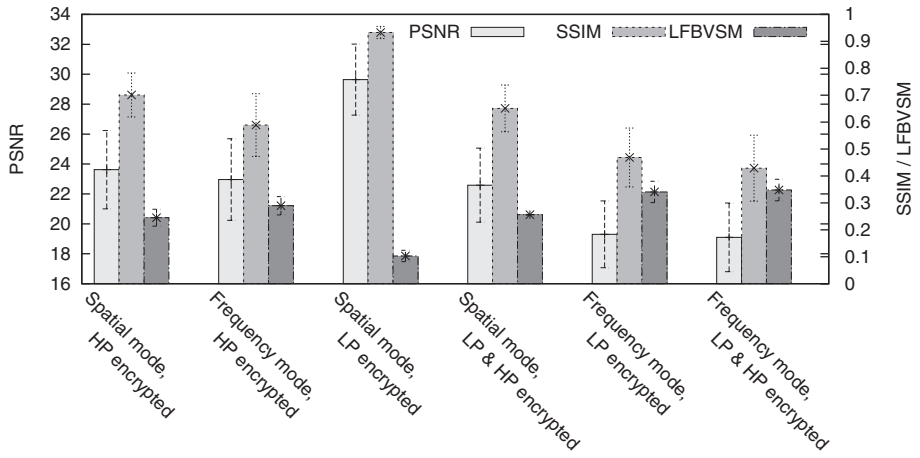

Figure 16 Mean values and standard deviation of objective image metrics for CSOP encrypted Kodak Image Database images.

and RLSE need to access (quantised) coefficients, thus are applied after the coefficient prediction stage. Entropy decoding is obviously required for those two techniques, which makes them already quite expensive in terms of computational cost when applied to JPEG XR bitstreams. Finally, TBE requires the original image to be accessible, thus, the entire process as depicted in Figure 1 needs to be conducted to apply TBE, which seems to be inacceptable for an actual application when applied to given images or bitstreams.

4.3 Computational cost

From the aspect of computational cost for the encryption process itself (which is the only additional cost in case of compression-integrated application), there are significant differences among the techniques discussed. Most specific media encryption schemes have been designed to limit computational cost, thus trade off security and compression efficiency for higher execution speed. On the other hand, classical cryptographic ciphers can (and should) be used for EEFB and SBE, which makes these techniques more costly compared to the remaining ones. SBE is very attractive due to the encryption of a small share of data only, which can be even reduced if not all coefficient signs are considered. In case EEFB protects all coefficient data, it is even slightly more demanding as conventional encryption. CSOP uses permutation as its cryptographic engine, which is well known to be of lightweight nature. RLSE seems to be even more lightweight by simply requiring a single arithmetic operation per coefficient; however, contrasting to CSOP, each coefficient is processed. So it finally depends on the hardware if CSOP or RLSE is more efficient. Finally, TBE comes at virtually no additional computational cost (only in case of compression-integrated application as we have seen above).

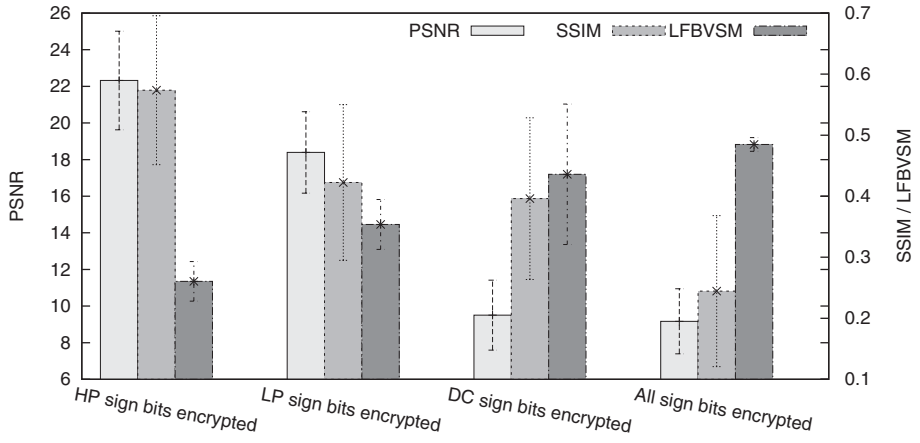

Figure 17 Mean values of the objective image metrics for the sign-encrypted Kodak IDB images.

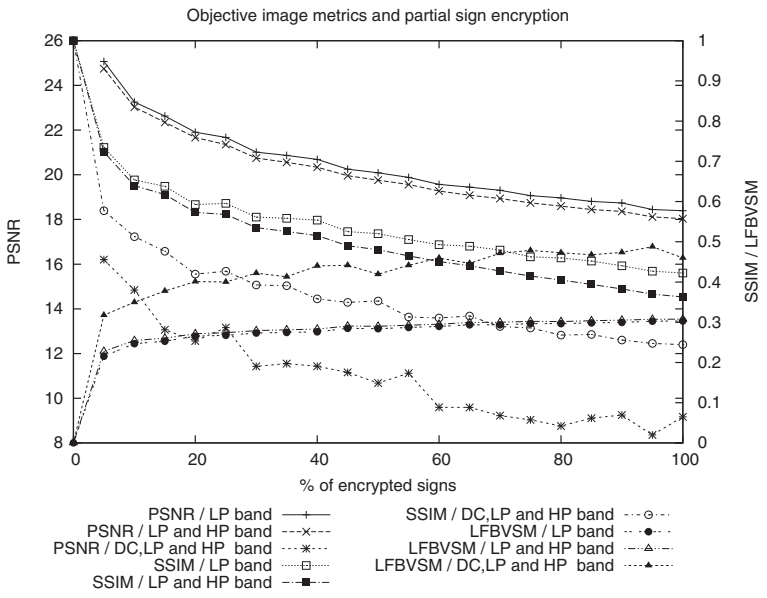

Figure 18 Mean values of the objective image metrics on the Kodak IDB encrypted with partial SBE. The *x*-axis shows the percentage of randomly selected and encrypted sign bits.

4.4 Robustness towards data manipulation

The robustness of encrypted multimedia data with respect to certain data manipulations is of interest in many application scenarios. While the conventional encryption approach (which is the encryption of the entire JPEG XR data with a cryptographically strong cipher in our context) of course does not offer such robustness properties due to the intrinsic properties of the employed ciphers, certain multimedia encryption techniques have been proven to provide this property to some extent. For example, we were able to demonstrate transmission error robustness and compression robustness for a class of chaos-based encryption techniques when applied to image data [15]. Contrasting to this prior work, we do not consider the encryption of plain image data here but we deal with format-compliant encryption schemes for a particular compression format, i.e. JPEG XR. Thus, it does not make sense to consider robustness against image-based manipulations like e.g. compression or noise insertion since for investigating this, the encrypted JPEG

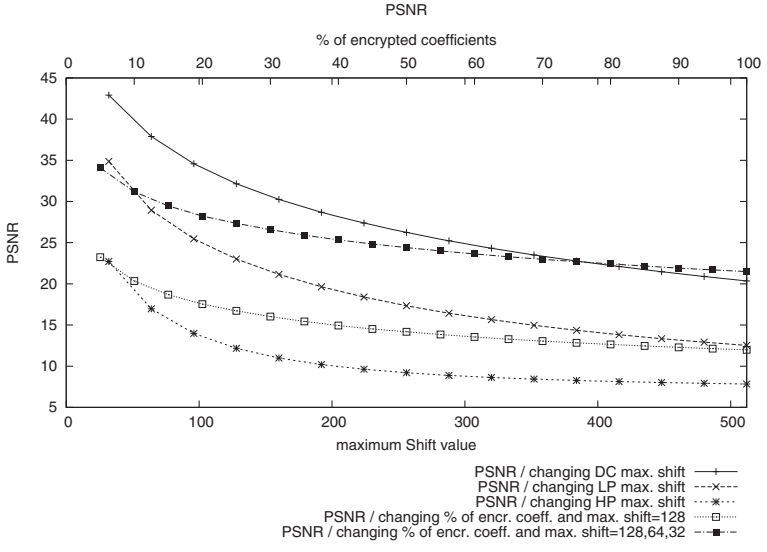

Figure 19 Averaged PSNR values of the random level shift encrypted Kodak IDB images.

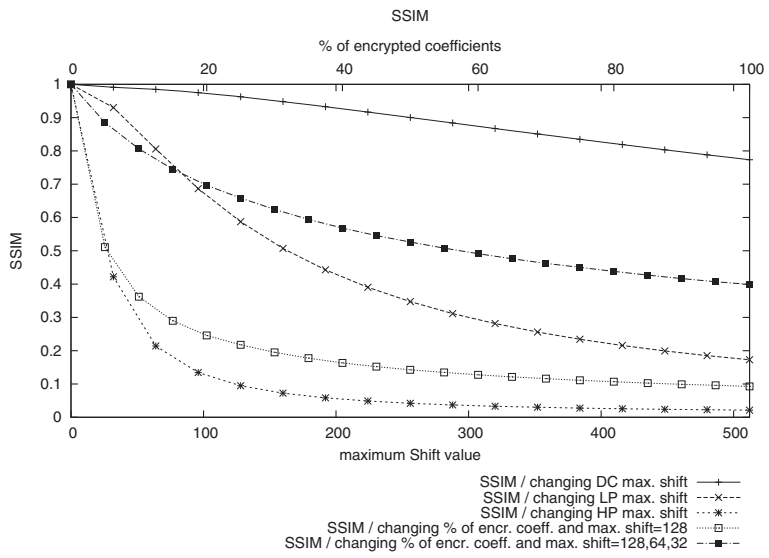

Figure 20 **Averaged SSIM values of the random level shift encrypted Kodak IDB images.**

XR file would have to be converted to plain image data (decoded - it is unclear if with correct or incorrect key in case of TBE) and the resulting image manipulated and subsequently reencoded (with identical encoding parameters). This does not really correspond to a realistic application context, and in addition to that, lossy recompression adds an additional layer of data degradation. On the other hand, robustness against transmission or storage errors is highly relevant, since the encrypted JPEG XR data of course may suffer from this type of degradation due to transmission errors of defects of storage media.

The JPEG XR standard offers only limited robustness against transmission errors itself, without considering any additional encryption. Bit errors in the VLC coded part of the code stream lead to a loss of synchronisation due to the generation of non-compliant codewords and the adaptive codebook selection, like described for VLC-based encryption. Also, bit errors in other parts of the code stream will likely result in all kind of decoding errors. Adding encryption certainly does not improve the situation. While TBE, CSOP, and RLSE transmission errors do not propagate to other parts of the bitstream during encryption (thus, transmission robustness is determined by the JPEG XR

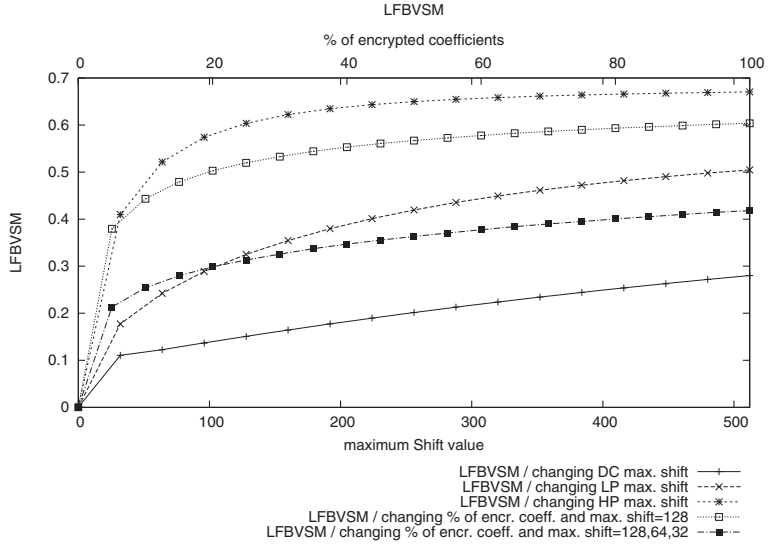

Figure 21 **Averaged LFBVSM values of the random level shift encrypted Kodak IDB images.**

Table 2 Mean values of objective image metrics for TBE-protected Lena image found in Figure 14

Metric	PCT decoded	DCT decoded
PSNR	17.80 dB	23.65 dB
SSIM	0.28	0.58
LFBVSM	0.52	0.44

data itself for these schemes), the application of cryptographically strong ciphers in SBE and EEFB leads to a propagation of transmission errors to other encrypted parts of the bitstream when these error-prone bits are decrypted. For SBE, this leads to encryption-like effects even when using the correct key for decryption (sign bits are set incorrectly), while for EEFB this causes even more JPEG XR decoding errors due to synchronisation loss. The extent of error propagation of course depends on the employed actual encryption mode of the ciphers in SBE and EEFB.

Considering this analysis, it immediately gets clear that robustness against image-based manipulations like described above is also mainly determined by the corresponding property of JPEG XR itself for TBE, CSOP, and RLSE, while more problems arise when using SBE and EEFB.

5 Security analysis

Assessing and especially comparing the security of format-compliant encryption schemes is a difficult task, since security ultimately depends on the skills and level of information of an attacker. Some techniques exhibit specific security breaches due to the cryptographic primitives involved. Of course, the application of cryptographically strong ciphers does prevent unwanted attacks against possibly weak cryptographic schemes - in this sense, EEFB and SBE are unrivalled in terms of security. CSOP uses a simple permutation cipher, while RLSE is in fact a simple substitution cipher. While the single simple techniques provide either diffusion or confusion, respectively, they do not provide both properties, which would be required for a cryptographically strong scheme.

5.1 Key space size

The assessment of key space size is an important aspect with respect to the security of encryption schemes. A respective comparison of the discussed techniques can be found in Table 4. The key space of many methods is content dependent, e.g. it depends on the number of coefficients per block or number of overall non-zero coefficients.

Table 3 Averaged values when decoding TBE Kodak IDB images using lossless mode in PCT/DCT mode

Metric	Average	Deviation
PSNR	17.03/22.23 dB	1.84/1.62 dB
SSIM	0.21/0.56	0.09/0.13
LFBVSM	0.40/0.35	0.06/0.07

Key space size can become a problem for CSOP and TBE in case of small images and additionally for CSOP when the image material leads to a low number of non-zero coefficients (making $n_i!$ too small). In these cases, brute force attacks become feasible. The other techniques are not expected to run into problems concerning the size of the key space.

5.2 Attack resistance

Techniques protecting single coefficients or bands only are prone to the replacement attack [9] in which the encrypted data is simply replaced or ignored, thus revealing the unencrypted parts by getting rid of the noise introduced by decoding encrypted data parts. Furthermore, several types of side-channel attacks exist against such selective/partial encryption schemes [16], exploiting e.g. correlations between plaintext and ciphertext data parts resulting in *reconstruction attacks* [9] or error concealment attacks [17], where the latter type exploits format-specific error concealment strategies to recover plaintext data (here, also the availability of some plaintext parts in the encrypted data is a prerequisite for application). Therefore, all techniques being intrinsically or intentionally limited to specific bands (e.g. CSOP or RLSE of the LP band) exhibit in fact a lower level of (visual) security as being suggested by visual inspection or objective metrics applied to the encrypted data without attacks mounted.

Table 5 summarises the robustness of the discussed techniques against brute force attacks (BF-A), ciphertext-only attacks (CO-A), and known plaintext attacks (KP-A). If a technique allows the application of partial/selective encryption, also the intrinsic sensitivity against replacement attacks/error concealment attacks (EC-A) is documented.

The BF-A column basically summarises the findings with respect to key space size shown in Table 4.

CSOP, RLSE, and TBE are highly sensitive to the known plaintext attack (shown in the KP-A column). Once a single pair of plaintext and ciphertext bitstreams is given, the used permutation/substitution/transform primitive can be immediately identified. Thus, keys can only be used once for these techniques.

In the following, ciphertext-only attacks are discussed (shown in the CO-A column). This attack type has been demonstrated against MPEG coefficient permutation [18], which is applicable by analogy to the JPEG XR CSOP case. Coefficients are simply sorted according to their magnitude and inserted into the 2D coefficient matrix, which reveals medium quality images immediately. A

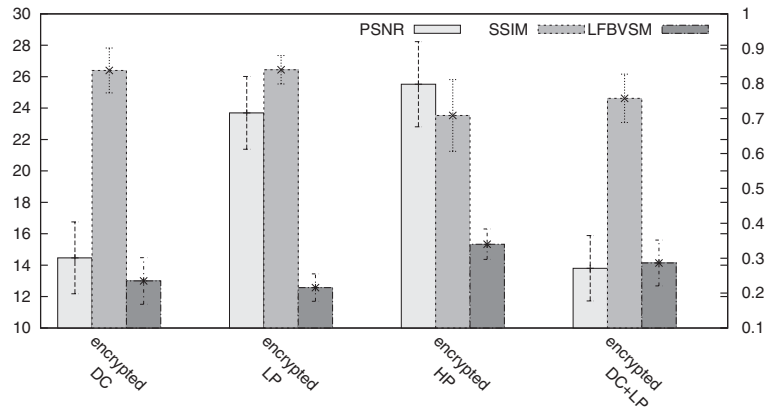

Figure 22 Averaged values and standard deviation of objective image metrics for EEFB on Kodak IDB images.

fundamental weakness and a corresponding ciphertext-only attack against encryption using secret transform domains has been described [19], thus also affecting TBE. The idea is to optimise image smoothness measures locally and exploiting the fact that the two coordinate axes can be attacked independently thereby reducing key space considerably. Also SBE can be attacked by ciphertext only, based on the analysis of the colour clipping effect as follows. Colour values become clipped in SBE due to a value overflow during inverse transformation when not using the valid key. Due to this effect, pixel values are set to their minimum or maximum value. The result can be seen in Figure 8 where completely white spots indicate an 8-b overflow of all three colour bands when transforming the coefficients back into YUV/RGB. This behaviour may be used to recover the sign bits for the affected regions of the image in an automated fashion. However, this attack strategy has some limitations since it only allows to recover signs of transform blocks where a clipping effect can be observed. Additionally, because of the prediction used in JPEG XR, the colour clipping effect of a manipulated sign bit may not appear immediately but is passed on to a subsequent transform block. This significantly increases the complexity of the attack.

Table 4 Comparison of the key space of presented encryption methods

Method	Key space
Coefficient scan order permutation (CSOP)	$Mn_i!$
Sign bits encryption (SBE)	k
Random level shift encryption (RLSE)	S^{n_t}
Transform based encryption (TBE)	16^M
Encryption of frequency bands (EEFB)	k

k is the key space of an employed cryptographically strong cipher.

The table illustrates that with respect to security, only EEFB exhibits robustness against all considered attack types (except for EC-A which are only applicable in case EEFB is applied as a partial/selective encryption scheme), while all other approaches show at least one or even multiple weaknesses.

6 Conclusions

There is no 'perfect' solution for format-compliant JPEG XR encryption. The discussed techniques differ significantly in terms of computational cost, security, range of applicability, compression impact, and adherence to format compliance. Properties which are most important can only be determined according to a specific application context.

CSOP, SBE, RLSE, and TBE achieve format compliance according to the definition. However, CSOP and TBE are not qualified at all for real-world application due to significant security problems, their high computational cost when being applied to bitstreams due to required decoding, and the inacceptable impact on filesize/quality. Also, RLSE is weak in terms of security and costly in terms of computation when being applied to bitstreams. A significant impact on coding performance is observed for most

Table 5 Robustness against various types of attacks

Method vs. attack	BF-A	KP-A	CO-A	EC-A
CSOP	✗	✗✗	✗✗	✗
SBE	✓	✓	✗	✗
RLSE	✓	✗✗	✓	✗
TBE	✗	✗✗	✗✗	✓
EEFB	✓	✓	✓	✗

✓ denotes robustness against the attack, ✗ denotes partial sensitivity against the attack, resulting in e.g. a reduced complexity of a subsequent brute force attack or an approximative image reconstruction, and ✗✗ denotes a high sensitivity against the attack.

settings as well. SBE can be an option if content security is not required due to its low computational demand (also when applied to bitstreams) and has no impact on compression performance. However, there is an attack based on observed colour clipping effects, which can endanger protection locally even though strong ciphers are used for encryption.

On the other hand, EEFB exhibits many desired properties. First, it requires minimal parsing effort when being applied to bitstreams. Second, this method is the best method in terms of cryptographic security and has a negligible impact on code-stream size. From the application perspective, this method qualifies for all realistic scenarios (with the small restriction that only a low number of degradation settings is available without any intermediate stages). However, when applied in compression-integrated manner, EEFB exhibits the highest computational cost of all techniques considered due to the application of a cryptographically strong cipher. EEFB does not exactly adhere to the definition of format compliance since the bitstream contains encrypted data parts not covered by the standard. On the other hand, JPEG XR files encrypted with EEFB can be decoded with the reference software or any other software following the required specifications. Thus, in applications where the aim is only to decode, visualise, or process the decoded data, this weak form of format compliance is equivalent to the usual definition and therefore, EEFB is the perfect solution to achieve format-compliant encryption. For other application types, it needs to be assessed, whether this weaker form of format compliance suffices the application requirements.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
This work has been partially supported by the Austrian Science Fund, FWF TRP project L554.

Received: 16 October 2013 Accepted: 7 March 2014
Published: 4 April 2014

References

1. L Tang, Methods for encrypting and decrypting MPEG video data efficiently, in *Proceedings of the ACM Multimedia 1996* (Boston, USA, November 18-22, 1996), pp. 219–229
2. B Bhargava, C Shi, Y Wang, MPEG video encryption algorithms. Multimed. Tool. Appl. **24**(1), 57–79 (2004)
3. J Wen, M Severa, W Zeng, M Luttrell, W Jin, A format-compliant configurable encryption framework for access control of video. IEEE Trans. Circ. Syst. Video Tech. **12**(6), 545–557 (2002)
4. S S-KA Yeung, B Zhu, Zeng, Partial video encryption based on alternating transforms. IEEE Signal Process Lett. **16**(10), 893–896 (2009)
5. S-KA Yeung, S Zhu, B Zeng, Partial video encryption based on alternating integer transforms, in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)* (Paris, 30 May to 2 Jun 2010), pp. 833–836
6. F Dufaux, GJ Sullivan, T Ebrahimi, The JPEG XR image coding standard. IEEE Signal Process Mag. **26**(6), 195–199 (2009)
7. H Sohn, W DeNeeve, YM Ro, Region-of-interest scrambling for scalable surveillance video using JPEG XR, in *ACM Multimedia 2009* Beijing, 19–22 October 2009, pp. 861–864
8. H Sohn, W De Neve, YM Ro, Privacy protection in video surveillance systems: analysis of subband-adaptive scrambling in JPEG XR. IEEE Trans. Circ. Syst. Video Tech. **21**(2), 170–177 (2011)
9. A Uhl, A Pommer, *Image and Video Encryption. From Digital Rights Management to Secured Personal Communication*, Advances in Information Security, vol. 15. (Springer, Berlin, Heidelberg, New York, Tokyo, 2005)
10. ITU-T, T.832: Information technology - JPEG XR image coding system - Image coding specification. 2009, pp. 43
11. J Hämmerle-Uhl, S Jenisch, A Uhl, Format compliant RoI encryption of JPEG XR bitstreams based on tiling, in *Proceedings of the 21st European Signal Processing Conference, EUSIPCO '13* (Marrakech, Morocco, September 2013), pp. 9–13
12. Z Wang, AC Bovik, HR Sheikh, EP Simoncelli, Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
13. L Tong, F Dai, Y Zhang, J Li, Visual security evaluation for video encryption, in *Proceedings of the International Conference on Multimedia, MM '10* (ACM New York, 2010), pp. 835–838
14. R Franzen, Kodak Lossless True Color Image Suite. accessed 18 February 2012, http://r0k.us/graphics/kodak/
15. M Gschwandtner, A Uhl, P Wild, Transmission error and compression robustness of 2D chaotic map image encryption schemes. EURASIP J. Inform. Secur. **2007**(Article ID 48179) (2007). doi:10.1155/2007/48179, 16 pages
16. A Said, Measuring the strength of partial encryption schemes, in *Proceedings of the IEEE International Conference on Image Processing (ICIP'05), volume 2* (Genoa, Italy September 11-14 2005), pp. 1126–1129
17. T Stütz, A Uhl, On JPEG2000 error concealment attacks, in *Advantages in Image and Video Technology: Proceedings of the 3rd Pacific-Rim Symposium on Image and Video Technology, PSIVT '09,* Tokyo 2009. Lecture notes in computer science (Springer Berlin, Heidelberg, New York, Tokyo, 2009), pp. 851–861
18. L Qiao, K Nahrstedt, Comparison of MPEG encryption algorithms. Int. J. Comput. Graph. (Special Issue on Data Security in Image Communication and Networks). **22**(3), 437–444 (1998)
19. D Engel, R Kutil, A Uhl, A symbolic transform attack on lightweight encryption based on wavelet filter parameterization, in *Proceedings of ACM Multimedia and Security Workshop, MM-SEC '06* (Geneva 26–27 September 2006), pp. 202–207

doi:10.1186/1687-417X-2014-6
Cite this article as: Jenisch and Uhl: **A detailed evaluation of format-compliant encryption methods for JPEG XR-compressed images.** *EURASIP Journal on Information Security* 2014 **2014**:6.