EURASIP Journal on Information Security
a SpringerOpen Journal

RESEARCH Open Access

Assessing JPEG2000 encryption with key-dependent wavelet packets

Dominik Engel[1,2], Thomas Stütz[1,2*] and Andreas Uhl[2]

Abstract

We analyze and discuss encryption schemes for JPEG2000 based on the wavelet packet transform with a key-dependent subband structure. These schemes have been assumed to reduce the runtime complexity of encryption and compression. In addition to this "lightweight" nature, other advantages like encrypted domain signal processing have been reported. We systematically analyze encryption approaches based on key-dependent subband structures in terms of their impact on compression performance, their computational complexity and the level of security they provide as compared to more classical techniques. Furthermore, we analyze the prerequisites and settings in which the previously reported advantages actually hold and in which settings no advantages can be observed. As a final outcome it has to be stated that the compression integrated encryption approach based on the idea of secret wavelet packets can not be recommended.

1. Introduction

For securing multimedia data–like any other type of data–full encryption with a state-of-the-art cipher, is the most secure option. However, in the area of multimedia many applications do not require the level of security this option provides, and seek a trade-off in security to enable other requirements, including low processing demands, retaining bitstream compliance and scalability, and the support for increased functionality, such as, transparent encryption [1] and region of interest (ROI) encryption, or signal processing in the encrypted domain (adaptation, searching, watermarking, ...).

JPEG2000 is the most recent and comprehensive suite of standards for scalable coding of visual data [2,3]. JPEG2000 filled areas of application that JPEG could not provide for, especially where applications require a scalable representation of the visual data. Recently JPEG2000 has evolved into the format of choice for many specialized and high end applications. For example, the Digital Cinema Initiative (DCI), an entity created by seven major motion picture studios, has adopted JPEG2000 as the compression standard in their specification for a unified Digital Cinema System [4]. As a second example, in 2002, the Digital Imaging and Communications in Medicine (DICOM) committee approved the final text of DICOM Supplement 61, marking the inclusion of Part 1 of JPEG2000 in DICOM (ISO 12052). Furthermore, in the ISO/IEC 19794 standard on Biometric Data Interchange Formats JPEG2000 is included for lossy compression, in the most recently published version (ISO/IEC FDIS 19794-6 as of August 2010) as the only format for iris image data.

Security techniques specifically tailored to the needs of scalable representation in general and JPEG2000 in particular have been proposed recently, e.g., [5-10]. An overview and discussion of the proposed approaches in the context of JPEG2000 can be found in [11]. JPEG2000 security is discussed in JPEG2000 Part 8 [12,13]. This part has the title "Secure JPEG 2000" and is referred to as JPSEC. It "intends to provide tools and solutions in terms of specifications that allow applications to generate, consume, and exchange Secure JPEG 2000 codestreams" (p. vi). JPSEC extends the codestream syntax to allow parts which are created by security tools, e.g., cipher or authentication tools. Encryption is implemented with conventional ciphers, e.g., AES.

The approaches discussed in this article perform encryption by constructing a secret transform domain. The principal idea of such schemes is that without the key the transform coefficients cannot be interpreted or decoded and therefore no access to the source material is possible (or only at a very low quality). Other than with bitstream-oriented methods, which operate on a final

* Correspondence: tstuetz@cosy.sbg.ac.at
[1]University of Applied Sciences, Urstein Sued 1, 5412 Puch/Salzburg Austria
Full list of author information is available at the end of the article

coded media bitstream, these methods apply encryption integrated with compression. In terms of applicability they are therefore restricted to scenarios where the final media bitstream is not yet available (video conferencing, live streaming, photo storage, transmission, and storage of surveillance data etc.).

Transparent encryption was introduced in the context of TV broadcasting [1,14] and denotes encryption schemes for which public access is granted for a preview image, i.e., anyone can decode an image of reduced quality from the encrypted stream, even without the key data. The difference to other media encryption schemes that guarantee a certain degree of distortion is that the preview image has to be of a (specified) minimum quality, i.e., apart from the *security requirement*, there is also a *quality requirement* [10,15]. Broadcasting applications, for example, can benefit from transparent encryption, as they, rather than preventing unauthorized viewers from receiving and watching their content completely, aim at promoting a contract with non-paying watchers, for whom the availability of a preview version (in lower quality) may serve as an incentive to pay for the full quality version. The reason for considering transparent encryption as target application scenario is that it has been shown [16], that the lowest resolution contained in a JPEG2000 file encrypted using the techniques discussed in this article can always be decoded by an attacker. This makes the approaches suitable for transparent encryption only, but prevents usage in applications requiring a higher degree of confidentiality.

In [17] the authors suggest to use secret Fourier-based transforms for the encryption of visual data. Other proposals in the area of lightweight encryption [18] propose the encryption of the filter choice used for a wavelet decomposition. However, this suggestion remains vague and is not supported by any experiments, while [19,20] propose encrypting the orthogonal filterbanks used for an non-stationary multi-resolution analysis (NSMRA) decomposition. The use of concealed biorthogonal parametrized wavelet filters for lightweight encryption is proposed by [21]. The use of key-dependent wavelet packet decompositions is proposed first by [22,23]. The latter study [23] evaluates encryption based on key-dependent subband structures in a zerotree-based wavelet codec. Parametrized wavelet filters have been employed for JPEG2000 lightweight encryption by [24,25], however, this approach was shown to be insecure in later study [26]. The scrambling of discrete cosine transform (DCT) and discrete wavelet transform (DWT) coefficients is proposed in [27]. Recently secret DCT-based transforms have been proposed for image and video encryption [28].

In the context of JPEG2000, the degrees of freedom in the wavelet transform are a prime candidate for constructing a secret transform domain. JPEG2000, Part 2, allows the definition of custom wavelet filters and user-defined isotropic and anisotropic wavelet packet subband structures [29]. Exemplary wavelet packets are shown in Figure 1.

Key-dependent wavelet packets in JPEG2000 have been proposed for a lightweight encryption scheme in earlier studies [16,30,31]. This approach is in the focus of interest in this study. The suggested scheme can be seen as a form of header encryption, as only the information pertaining to the transform domain needs to be encrypted, the rest of the data remains in plaintext. This approach has the advantage that only the parameters of the secret transform domain need to be kept secret. Therefore the demands for the encryption stage are minimal as compared to a more traditional, bitstream-oriented encryption approach [16]. Due to the shift in complexity from actual encryption to the compression pipeline, the scheme has been termed "lightweight". An overview of the two different systems KDWP encryption and conventional encryption is shown in Figure 2.

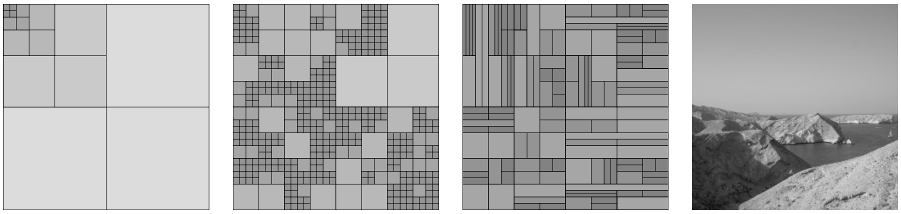
In this article, we evaluate, analyze, and discuss earlier proposed JPEG2000 encryption techniques that use key-dependent wavelet packets (KDWP) to establish a secret transform domain. Wavelet packets (WP) are introduced in Section 2.. We assess KDWP encryption with respect to the following three main properties of an image encryption scheme:

- *Compression impact*: KDWP encryption reduces compression performance, the actual decrease is evaluated in Section 3..
- *Computational demand*: The main argument for introducing the general concept of secret transform domains in encryption has always been an improved runtime performance, as the *conventional encryption* step is not needed. This assumption is in-depth evaluated and analyzed in Section 4..
- *Security*: The security of KDWP is in-depth analyzed in Section 5.. Special care has been taken to employ the suitable notion of security for multimedia encryption.

In Section 6. we combine the individual assessments of the previous sections and in-depth discuss other potential advantages of KDWP. Another potential advantage of KDWP encryption is the capability of performing signal processing operations on the protected data ("encrypted domain processing"), which are compared to the operations possible on a conventionally encrypted JPEG2000 bitstream. Final conclusion are drawn in Section 7..

2. Wavelet packets

The wavelet packet (WP) transform [32] generalizes the pyramidal wavelet transform. In the WP transform, apart from the approximation subband also the detail



(a) Pyramidal wavelet decomposition

(b) Isotropic WP

(c) Anisotropic WP

(d) Exemplary test image

Figure 1 Exemplary wavelet packet decompositions and test image.

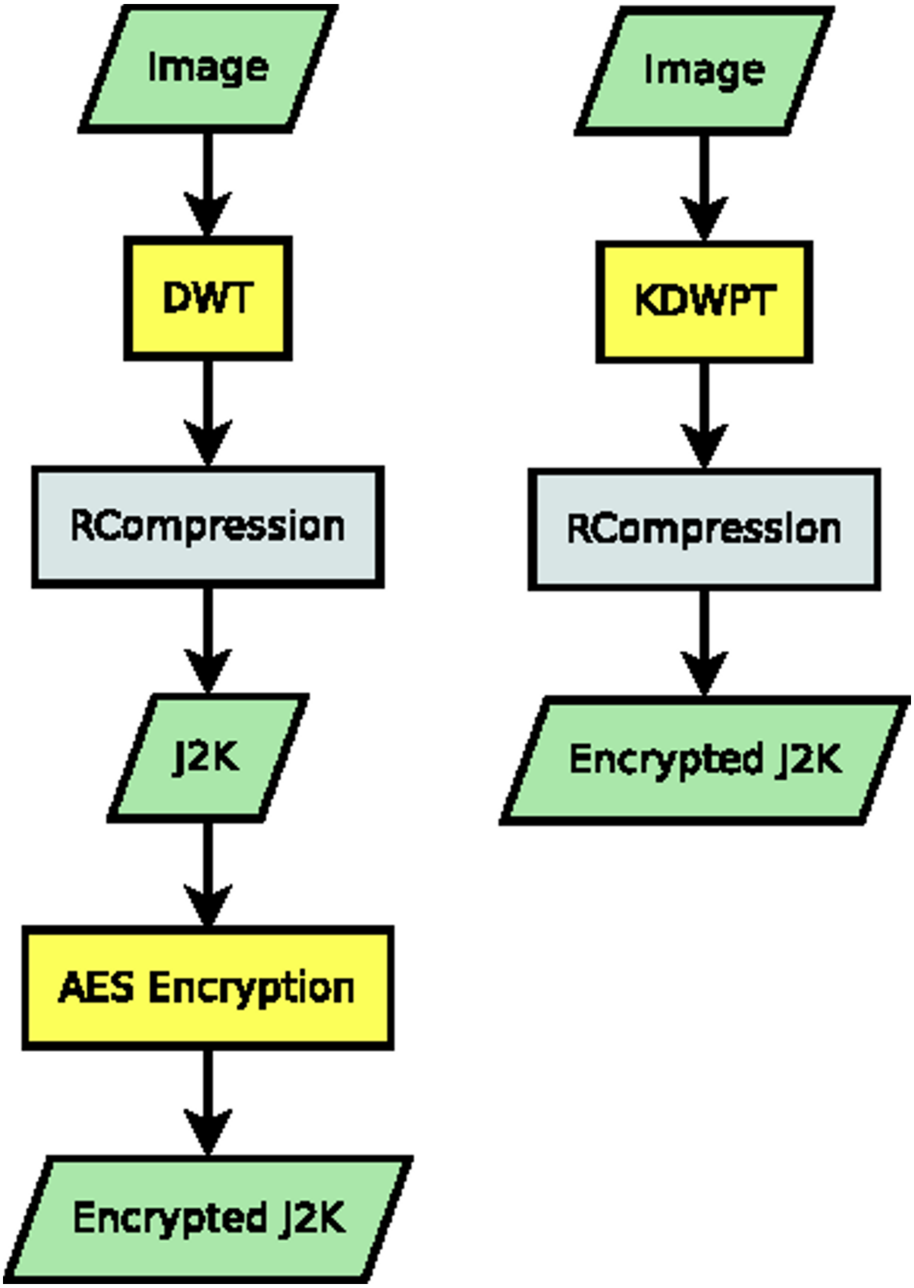


Figure 2 Conventional encryption versus KDWP.

subband can be decomposed; an example is shown in Figure 1. The WP can be adapted to take the properties of the image to be transformed into account, for example by using the best basis algorithm [32-36]. In this article we refer to each such a WP basis by the terms "WP subband structure", "decomposition structure" or briefly "WP".

The anisotropic WP transform is a generalization of the isotropic case: whereas in the latter, horizontal and vertical wavelet decomposition are always applied in pairs for each subband to be decomposed, this restriction is lifted for the anisotropic WP transform (for an example see Figure 1c).

Note that for the isotropic WP transform a single decomposition refers to both horizontal and vertical filtering and downsampling. For the anisotropic WP transform, a decompositions refers to filtering and downsampling in one direction (horizontal or vertical). Therefore, a decomposition depth of 2 g in the anisotropic case is comparable to a decomposition depth of g in the isotropic case.

2.1 WP in JPEG2000

Part 2 of the JPEG2000 standard [29] allows more decomposition structures than Part 1. Every subband resulting from a highpass filtering can be decomposed at most two more times (either horizontally, vertically or in both directions).

In order to maximize keyspace size for the proposed encryption scheme, we have implemented full support for arbitrary isotropic and anisotropic wavelet decomposition structures in JPEG2000, based on the JJ2000 reference implementation.[a] The source code for the implementation underlying all results in this paper can be downloaded from http://www.wavelab.at/sources.

2.2 Randomized generation of isotropic WPs

A WP can be derived by a sequence of random binary decomposition decisions. The seed s for the employed secure random number generator is the main parameter for the randomized generation of WP, i.e., comparable to the key in a symmetric crpyto system.

2.2.1 Uniform distribution

A uniformly distributed selection is achieved with the following randomized algorithm. For each subband (starting at the root subband, i.e., the entire image) it is randomly decided whether the subband is further decomposed. The probability of a decomposition of a subband depends on the number of sub-transforms within the subband, which is equivalent to $Q_{g\text{-}l}$ where g is the maximum decomposition depth and l is the level of the subband.

$$p_u(l) = 1 - \frac{1}{Q_{g-l}}, \quad Q_j = Q_{j-1}^4 + 1, \quad Q_0 = 1$$

Subsequently, this distribution is referred to *isouni.*

2.2.2 Compression-oriented distribution

The compression-friendly randomized algorithm for WP selection enforces the decomposition of the approximation subbands, for all other subbands a possible decomposition is determined by a layer-dependent probability, $p_c(l)$, which depends on two input parameters, the base value b and the change factor c, which serves as multiplier for the level l:

$$p_c(l) = 1 - \frac{b + cl}{2}$$

Only the parameters, b, c and the seed of the random number generator need to be encrypted. Subsequently, these distributions are denoted/abbreviated by *iso* and further classified into a constrained (the LL is always further decomposed) and an unconstrained case.

2.3 Randomized generation of anisotropic WPs

The main motivation to introduce anisotropic WPs in the context of lightweight encryption is a significant increase in keyspace size [30,31]. This increase is due to the fact that the anisotropic transform has substantially more WP for a comparable maximum decomposition depth. The number anisotropic WP is given by the following recursion:

$$R_j = 1 + 2R_{j-1}^2 - R_{j-2}^4, \quad R_{-2} = R_{-1} = 0$$

Even more than in the case of isotropic WPs, there are anisotropic WP decompositions that are ill-suited for energy compaction. The compression-oriented selection method tries to eliminate these subband structures.

2.3.1 Uniform distribution

We use the case distinction introduced by [37] to construct a uniform distribution for the selection of a random subband structure: the probability for any case to be chosen is the ratio of the number of subband structures contained in the case to the total number of subband structures. Subsequently, this distribution is often denoted/abbreviated by *anisouni.*

2.3.2 Compression-oriented distribution

The basic algorithm for the compression-oriented generation of randomized anisotropic WPs is similar to the isotropic case, only the direction of the decomposition (vertical or horizontal) is additionally chosen at random. However, constraining the degree of anisotropy is necessary in order to prevent subbands from being decomposed excessively in a single direction, as, especially in the case of the approximation subband, this would lead to inferior energy compaction in the transform domain and thus to inferior compression results. The parameter q is used to restrict the maximum degree of anisotropy for the approximation subband. For the degree of anisotropy Υ of a subband we use the following definition:

$$\Upsilon(h, v) = v - h \tag{1}$$

where h and v are the decomposition depths in horizontal and vertical direction, respectively. If at any node during the randomized generation of an anisotropic WP subband structure, decomposition of the subband at this node in the randomly chosen direction would result in the degree of anisotropy exceeding the maximum degree of anisotropy, the direction of the decomposition is changed. The degree of anisotropy for the approximation and detail subbands influence both, compression performance and keyspace size.

The other parameters are used in the same way as in the isotropic case: The base value b sets the basic probability of decomposition, the change factor c alters this base probability depending on the current decomposition level l.

3. Evaluation of compression performance

In previous study [16,30] parameter settings for the compression-oriented distribution have been determined for a small number of test images. For the isotropic wavelet packet transform it is proposed to force a maximum decomposition depth for the approximation subband. For the anisotropic wavelet packet transform, in addition to forcing a maximum decomposition depth, the maximum degree of anisotropy for the approximation subband needs to be restricted to preserve compression performance.

The parameters proposed by [16,30] were obtained empirically by a number of experiments. A large number of different parameter settings were used, but only on three test images. The parameters that were obtained for these three test images are a base value b of 0.25 and a change factor c of 0.1. For the isotropic case the global decomposition depth g has been set to 5. For the anisotropic case g has been set to 10 and the maximal degree of anisotropy of the approximation subband q has been set to 1. Recall that we follow the convention to give the decomposition depth of the isotropic wavelet packet transform in pairs of (horizontal and vertical) decompositions, whereas in the anisotropic case each (horizontal or vertical) decomposition step is counted separately.

We use these parameters in our experiments and evaluate their performance for a larger set of test images. We verify the compression performance of the compression-oriented selection method by an empirical study based on an extended set of images. For this purpose we use a set of 100 gray-scale images of 512 × 512 pixels (taken with four different camera models).

We use five different bitrates: 0.125, 0.25, 0.5, 1, and 2 bpp. For each of the test images we performed the following JPEG2000 compression tests at each of these bitrates:

- Pyramidal (1 subband structure, level 5),
- Isouni: random isotropic wavelet packets drawn according to the uniform distribution with $g = 5$ (100 randomly selected subband structures),
- Iso (constrained): random isotropic wavelet packets drawn according to the compression-oriented distribution with $g = 5$, $b = 0.25$, and $c = 0.1$ (100 randomly selected subband structures),
- Anisouni: random anisotropic wavelet packets drawn according to the uniform distribution with $g = 10$ (100 randomly selected subband structures), and
- Aniso (constrained): random isotropic wavelet packets drawn according to the compression-oriented distribution with $g = 10$, $b = 0.25$, $c = 0.1$, and $q = 1$ (100 randomly selected subband structures).

To ensure comparability the same seeds (and therefore the same decomposition structures) were chosen for each image at each of the five different rates. The standard CDF 9/7 biorthogonal wavelet was used for transformation in all experiments. The results of our empirical study are summarized for the five categories and all bitrates in Figure 3.

For the compression-oriented setup, the loss in compression performance is smaller for the anisotropic randomized decomposition method (below 1dB). Due to the fact that randomized anisotropic wavelet packets require fewer decompositions for the same keyspace size, the compression performance achieved in the anisotropic setup is superior to the isotropic setup. For the set of natural test images the pyramidal decomposition remains the setup with the best compression performance. In part this is due to the overhead in header data that is introduced in the JPEG2000 bitstream by increasing the number of subbands (as is usually the case in KDWP). Table 1 shows the average ratio of header data to packet data for all test images at different bitrates. As the header size is less affected by bitrate it can be seen that the ratio increases when the bitrate decreases and can make up a substantial part of the bitstream.

As regards the difference between uniform and compression-oriented selection, it can be seen that the compression performance of the latter is above the compression performance of the former. The difference is more evident for the anisotropic case, for which a predominant decomposition of the approximation subband in a single direction, which leads to inferior energy compaction for natural images, is possible. Restricting the

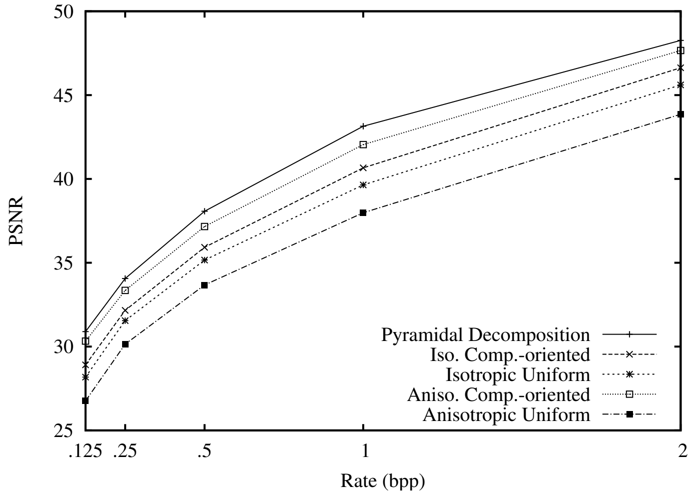


Figure 3 Empirical results: average compression performance (100 images).

maximum degree of anisotropy for the approximation subband in the compression-oriented selection leads to a compression performance that is closer to the pyramidal decomposition.

From an compression performance point of view, the compression-oriented distributions are favorable, although, even these distributions considerably decrease compression efficiency.

4. Computational complexity

In order to compare the computational complexity of KDWP encryption compared to conventional compression and encryption it is sufficient to determine the difference. The main difference between KDWP encryption and conventional compression and encryption is the transform stage, a random WP is chosen for KDWP encryption, while in a conventional encryption approach the pyramidal decomposition is employed. In the commonly applied PCRDO coding approach (post-compression-rate-distortion optimization) the complexity of the remaining compression (labeled “RCompression” in Figure 2) is almost completely independent of the bitrate and thus the difference in complexity is almost completely due to the difference in the transform stage. In the conventional approach there is an additional encryption process after compression, which is linearly bit-rate-dependent. First we present experimental performance results in Section 4..1 and complement the findings with a theoretical performance analysis in Section 4..2.

4.1 Experimental performance evaluation

The experimental results have been obtained using our JJ2000-based implementation. The test set consisted of 100 grey-scale images with a resolution of 512 × 512. Both the implementation and the test set will be publicly available at http://www.wavelab.at/sources/. The tests were conducted on an Intel Core 2 CPU 6700 @2.66 GHz. Table 2 summarizes the results for the pyramidal decomposition and the different selection schemes, the uniform distribution ("isouni”) and the compression-oriented distribution ("isouni”). The results are averages of hundred trials. Additionally we need to determine the complexity of AES encryption, which is given in Table 3 [11]. These results enable us to determine the complexity in dependence of a varying bit-rate, which is shown in Figure 4. The bit-rate in bpp (bit per pixel) is plotted against the complexity given in average processing time for in image in seconds. Clearly the additional complexity of AES encryption is negligible, compared to the complexity introduced by the KDWP. Furthermore the anisotropic

Table 1 Ratio of header data to packet data for different compression rates (16 quality layers)

Rate	pyramidal	iso	aniso
0.125	15.9%	32.8%	20.8%
0.25	10.0%	20.8%	15.1%
0.5	6.1%	13.6%	9.6%
1	3.7%	9.2%	5.9%
2	2.5%	6.4%	3.9%

Table 2 Average compression complexity with our implementation

structure	avg. *t*	avg. fps
pyramidal	0.643 s	1.55 fps
iso	1.005 s	1 fps
isouni	1.147 s	0.87 fps
aniso	0.726 s	1.34 fps
anisouni	1.891 s	0.53 fps

Table 3 Runtime performance of encryption routines

AES encryption			
Method	throughput	codestreams	with 2bpp
AES OFB	42.71 MB/s	0.0015 s	683.36 fps

uniform distribution performs worst by far, which is to some extend implementation-specific. Very anisotropic subbands are obviously not very efficiently dealt with our implementation.

However, one might argue that our Java based implementation does not reflect the state-of-the-art. One of the fastest implementations, the Kakadu implementation[b], even achieves 39.88 fps compared to 1.55 fps with our implementation (compression with the pyramidal decomposition, 2 bpp, 5 level wavelet decomposition and no quality layers), and thus is about 25 times faster. Therefore we also show results for an hypothetical optimized implementation that even outperforms Kakadu by a factor of 2, i.e., is 50 times faster than our implementation (see Figure 5). Nonetheless the basic assessment stays the same, no performance benefits with KDWP can be gained even with an optimized implementation.

The final question is whether it is at all possible to gain any performance benefits with KDWP (even with the most optimized implementation) compared to AES encryption. This question can only be answered with a theoretical analysis.

4.2 Theoretical performance analysis

We determine the computational complexities on a random access machine with an instruction set of basic operations, such as ^, &, +, *, and %.

As we are only interested in the difference of the complexities it is sufficient to determine the complexity of the pyramidal wavelet transform, the average WP transform, and AES encryption.

The wavelet transform stage consists of iterated filter operations, the number of filter operations depends on the WP and the number of pixels. For every input pixel and decomposition level two filter operations are required for the isotropic case. A single filter operation with an n-length filter consists of n multiplications, n additions, n MemReads and a single MemWrite, which yields 25 operations for $n = 8$ (JPEG2000's 9/7 irreversible filter).

In order to determine the average/expected complexity of the KDWP, we determine the expected average decomposition depth of the WP drawn from one of the two proposed selection schemes. The complexity analysis based on the average decomposition depth allows us to analyze the complexity of the WT and the KDWP independent of the image resolutions, as for every pixel the computational complexity is given by the average decomposition depth times the number of necessary operations for filtering. For the pyramidal wavelet decomposition with decomposition depth g, the average decomposition depth D_g^p is given by:

$$D_g^p = \sum_{i=0}^{g} 1/4^i =^{g\rightarrow\infty} 4/3 \approx 1.33333$$

The expected average decomposition depth for the uniform distribution on isotropic WP with a maximum decomposition depth g, D_g can be derived recursively:

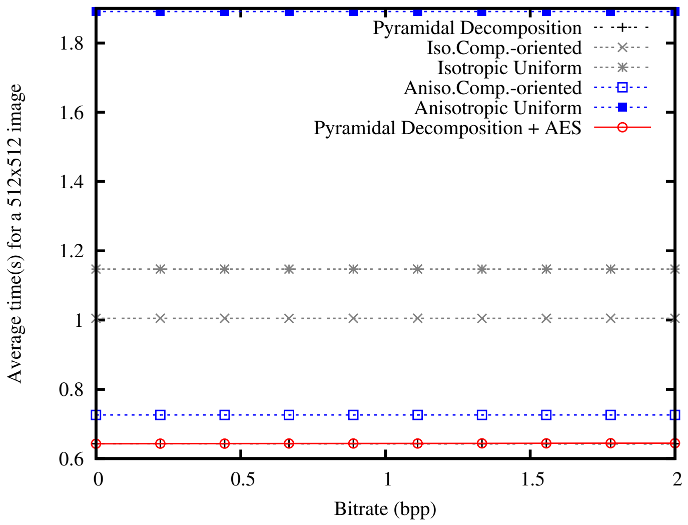


Figure 4 Comparison with our implementation.

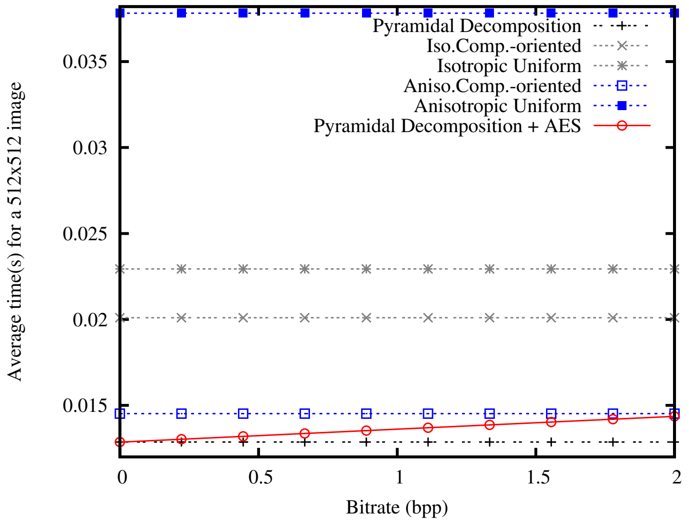


Figure 5 Comparison with an optimized implementation.

$$D_g = \frac{Q_g - 1}{Q_g}\left(D_{g-1} + 1\right), \quad D_0 = 0$$

$$g - 1 \leq D_g \leq g - 1/2, \quad D_{k+1} \approx k + 0.41174$$

The expected decomposition depth for the compression-oriented distribution without considering the forced decomposition of the approximation subbands is denoted D_g^c. For briefness we denote $p_c(l) = p_l$ and the probability of the converse event as $q_l = 1$ -p_l. $D_{l,g}^c$ gives the expected decomposition depth of a subband at level l with a maximum decomposition depth of g.

$$D_g^c = D_{0,g}^c, \quad D_{g,g}^c = 0, \quad D_{l,g}^c = \left(D_{l+1,g}^c + 1\right) p_l = \sum_{i=l}^{g-1} \prod_{k=l}^{i} p_k$$

If $c = 0$ and $1 - \frac{b}{2} < 1$, then:

$$D_g^c \overset{g \to \infty}{=} \frac{2}{b} - 1$$

If we take the enforced decomposition of approximation subbands into account (constrained case), we denote the expected average decomposition depth D_g^f.

$$D_g^f = \frac{g}{4^{g-1}} + \sum_{i=1}^{g-1} \frac{3}{4^i}\left(D_{i,g}^c + i\right)$$

If $c = 0$ and $0 < 1 - \frac{b}{2} < 1$, then:

$$D_g^f \overset{g \to \infty}{=} \frac{2}{b} + 1/3$$

Table 4 shows the average decomposition depths of D_g, D_g^c, and D_g^f for different maximum isotropic decomposition depths.

The expected decomposition depth for a uniform distribution on anisotropic WP is denoted D_g^a and can be determined by the following recursion:

$$\forall i \in \mathbb{N} : A_{-i} = 0, \quad A_g = 1 + 2A_{g-1}^2 - A_{g-2}^4$$

$$\forall i \in \mathbb{N} : D_{-i}^a = 0, \quad D_g^a = \frac{1}{A_g}\left(2A_{g-1}^2\left(D_{g-1}^a + 1\right) - A_{g-2}^4\left(D_{g-2}^a + 2\right)\right)$$

$$D_g^a \approx g - 1 + 0.5301$$

The expected decomposition depth for the compression-oriented distribution is denoted D_g^{ac}. $D_{l,g}^{ac}$ gives the expected decomposition depth of a subband at level l with a maximum decomposition depth of g.

$$D_g^{ac} = D_{0,g}^{ac}, \quad D_{g,g}^{ac} = 0, \quad D_{l,g}^{ac} = \left(D_{l+1,g}^{ac} + 1\right) p_l = \sum_{i=l}^{g-1} \prod_{k=l}^{i} p_k$$

If $c = 0$ and $0 < 1 - \frac{b}{2} < 1$, then:

$$D_g^{ac} \overset{g \to \infty}{=} \frac{2}{b} - 1$$

Table 5 shows the average decomposition depths of D_g^a, and D_g^{ac} for different maximum anisotropic decomposition depths.

Table 4 The expected average decomposition depth for isotropic WP distributions (b = 1/4, c = 0)

Iso. g	D_g	D_g^c	D_g^f
1	0.5	0.857	1
2	1.41176	1.6406	1.90625
3	2.41174	2.3105	2.70703
4	3.41174	2.89673	3.40967
5	4.41174	3.40946	4.02496

Table 5 The expected average decomposition depth for anisotropic WP distributions (*b* = 1/4, *c* = 0)

Aniso. *g*	D_g^a	D_g^{ac}
2	1.55556	1.64063
4	3.53125	2.89673
6	5.53020	3.85843
8	7.53014	4.59474
10	9.53014	5.15847

According to [38] and backed-up by our own analysis 352.625 operations are necessary for the encryption of a single byte with AES with a 128-bit key in CTR-mode.

The overall results are shown in Figure 6, the bitrate in bpp is plotted against the computational complexity in unit cost for the different approaches. Also the theoretical analysis backups the basic claim that no performance improvements can be gained from isotropic KDWP for the relevant bitrate ranges. Anisotropic uniform KDWP performs worst (similar to our experimental results). However, the difference to the other approaches is much smaller, clearly indicating that the JJ2000 implementation does not cope well with very anisotropic subbands. Only the anisotropic compression-oriented KDWP can theoretically perform slightly better for bit-rates in excess of 1.4 bpp. Note that even in the most optimized implementations this will not be achievable as AES has much more faster local data access, compared to the KDWP. Faster local data access is not reflected in our computational model.

5. Security evaluation

In order to assess the security of a multimedia encryption approach, we first need to define "security" of a multimedia cryptosystem more precisely. Conventional notions of security for cryptosystems require that the ciphertext does not leak any information (information-theoretic approach [39]) or any efficiently computable information (the approach of modern cryptography [40]) of the plaintext. This kind of security notions are also referred to as MP-security (message privacy) [41]. However this type of security is rarely met nor targeted by multimedia encryption schemes. Thus multimedia encryption is often analyzed with respect to a full message recovery, i.e., how hard is it for an attacker to reconstruct the (image) data. This type of security notion is referred to as MR-security (message recovery) [41]. However, a reconstruction of a multimedia datum (on the basis of the ciphertext) may have excellent quality and even be perceived as identical by a human observer, while the perfect recovery of the entire message remains impossible.

Thus in the context of multimedia encryption it is required to take the quality of a reconstruction (by an adversary on the basis of the ciphertext) into account. An adversary, who tries to break a multimedia encryption system, is successful if she can efficiently compute a "high quality" reconstruction of the original multimedia datum. Which quality constitutes a security threat highly depends on the targeted application scenario [11]. In [42] this multimedia-specific security notion is termed MQ-security (message quality), similar concepts can be found in the multimedia encryption literature [43,44].

5.1 Usual security analysis: key space

Commonly a multimedia encryption security analysis consists of counting the key space. The common conclusion is that if the key space is large enough the proposed approach is believed to be secure. The number of

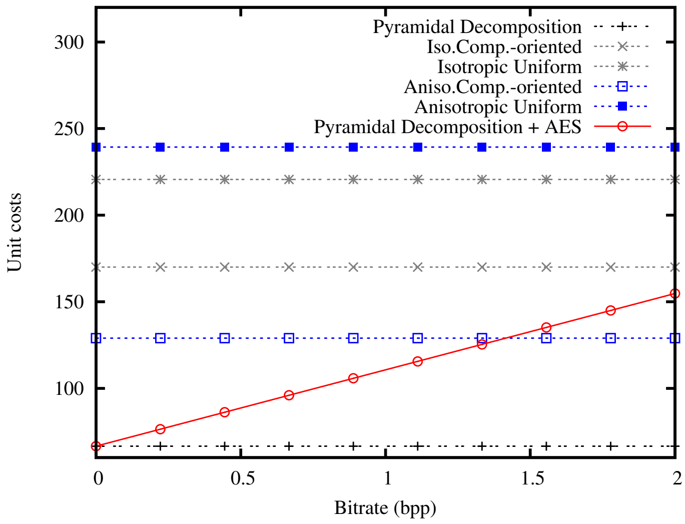


Figure 6 Theoretical comparison of computational complexity.

possible WP is huge even for moderate maximum decomposition depths (e.g., $2^{261.6}$ for isotropic $g = 5$, and $2^{1321.9}$ for an anisotropic depth of 10). Thus the common conclusion would be to consider the scheme secure. Additionally the quality of a reconstruction with a wrong key is often analyzed in literature. However, if we try to decode a JPEG2000 codestream with a wrong WP the decoder will not be able to decode an image, as the WP is required for the decoding. Information, such as the subband/codeblock size and the number of codeblocks in a subband, which is required to make sense of the coded data, is missing. Thus a naive attacker would need to test half of the possible WP before the correct is identified (on average). A futile approach given the number of WP!

5.2 Improved security analysis: entropy

We have presented two basic KDWP selection schemes, namely uniform and compression-oriented. So far the compression-oriented scheme shows advantages, both in terms of compression efficiency and runtime performance. However, the key space counting approach only reflects a security analysis if the WP are drawn according to a uniform distribution. How can we properly assess the security of the compression-oriented distribution? The appropriate measure is the entropy $H(X)$ of the compression-oriented distribution. An attacker needs to test $2^{H(X)-1}$ keys (WP) on average to find the correct key. The entropy $H(X)$ for the compression oriented distribution is a bit tricky to compute (see Appendix 1). Table 6 summarizes the results for previously proposed parameters and for the isotropic case.

Still the numbers are sufficiently large such that we have to conclude that KDWP in JPEG2000 are secure. Since entropy values in excess of state-of-the-art ciphers key set sizes (128 bit) can be considered secure, security is sufficient for $g > 5$ in the compression oriented case and for $g > 4$ in the case of uniform WP distribution.

Since the number of anisotropic WP is by far greater than the number of isotropic WP (for comparable maximum decomposition depths, remember an isotropic decomposition depth g corresponds to an anisotropic decomposition depth of 2 g), the anisotropic case has to be considered secure as well.

Table 6 Entropy of the isotropic compression-oriented distribution ($b = 1/4$, $c = 0$) and the uniform distribution

g	Entropy (iso)	Entropy (isouni)
2	2.4	4.1
3	9.1	16.3
4	32.4	65.4
5	114.0	261.6
6	399.5	1046.4

Note that this analysis holds for every scheme that strongly requires the WP for decoding. However, the key question now is whether this requirement is strong for JPEG2000 or whether it can be weakened by exploiting specifically tailored attacks.

In the following we will argue that such specifically-tailored attacks can be designed for KDWP with JPEG2000 and that they will be even more effective if only the permissible WP of JPEG2000 Part 2 are employed.

5.3 Specific attacks against KDWP in JPEG2000

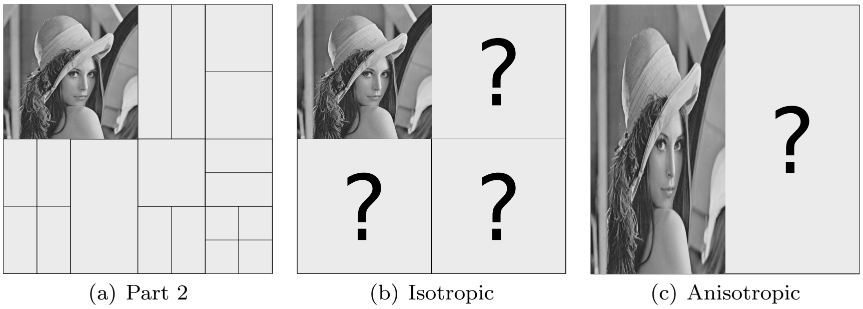
Due to JPEG2000 coding the LL subband can be reconstructed by an attacker with a special-purpose decoder. The LL subband can be decoded, because the coded LL subband data is located at the start of JPEG2000 file. Thus an attacker can decode a low-resolution image, a fact that has already been highlighted in previous study [16]. The previous conclusion has been that KDWP in JPEG2000 are only suitable for the application scenario of transparent encryption.

In this study we pose a new question: can an attacker go further and decode subsequent resolutions, i.e., images with an higher quality than targeted? In fact resolution only loosely corresponds to perceived quality, but as every subsequent improvement an attacker can achieve (by image enhancement operations) starts from a previously deciphered lower resolution it makes sense to use resolution as sole quality indicator within the scope of this work. The quality of an attacked image is determined as the fraction of original resolution divided by the obtained resolution (given in the number of pixels).

Thus the attacker's problem is: given access to a low-resolution image how hard is it to obtain/decode the next resolution? Figure 7 illustrates the problem for isotropic and anisotropic decompositions.

Therefore, we first need to answer whether the next resolution can be decoded from the codestream (independently of the higher resolutions). This is the case in the coding framework of JPEG2000. A resolution can be decoded independently from the remaining higher resolutions (definitely for resolution progression and at least at the lowest quality for layer progression and always if SOP and EPH markers are employed which signal packet borders).

The next question is whether it is decidable that the employed subband decomposition structure is the correct one. It is also highly likely that it can be decided whether the correct decomposition structure has been employed in the decoding of a resolution: Firstly, the wavelet resolutions are not independent, i.e., statistical cross-resolution dependencies are highly likely to identify the correct decomposition. Secondly, the codestream syntax and semantics must also be met while decoding



(a) Part 2 (b) Isotropic (c) Anisotropic

Figure 7 Attack against a subsequent resolution.

with a subband decomposition structure, i.e., decoding errors clearly indicate an incorrect decomposition structure.

Thus the decomposition structure of a resolution can be determined independently of the higher resolutions in JPEG2000.

Now, how hard is it for an adversary to decode a certain resolution? We first discuss the standardized case of JPEG2000 Part 2.

5.3.1 JPEG2000 Part 2

In JPEG2000 Part 2 a high frequency subband may only be decomposed two more times (either isotropic or anisotropic). In the case of isotropic WP, every next resolution may only be secured by $Q_2^3 = 17^3 = 4913$ possibilities. In the case of anisotropic WP, a subband has only $R_2 = 18$ possibilities to be further decomposed. Thus the restrictions of Part 2 and both distinct cases do not offer security for each subsequent resolution. An attacker would need only 4913/2 or 18/2 trials on average to decode each subsequent resolution.

However, JPEG2000 Part 2 allows to specify either horizontal, vertical or horizontal and vertical decomposition for a subband, i.e., there are four different subband structures for a single subband (no decomposition, vertical, horizontal, or both) in one step. The low resolution subband can have three high frequency subbands (HL, LH, HH) or one high frequency subband (either horizontal or vertical). These high frequency subbands may be decomposed two more times, which yields $4^4 + 4^2 + 4^2 - 1 = 287$ different subband decomposition structures for such a subband. 4^4 is the number of possibilities after a decomposition in both directions and 4^2 is the number of possibilities after either a horizontal or a vertical decomposition, one WP is counted three times (thus -2) and no decomposition also has to be counted (+1). Figure 7a shows the case with three subbands. There are three cases:

- Three subbands (HL, LH, HH), which leads to $287^3 = 23639903$ possible WP.
- One vertical subband, which has 287 possible WP.
- One horizontal subband, which has 287 possible WP.

In summary there are about $2^{24.49}$ possible WP for a subsequent resolution in Part 2. Though less than 2^{24} checks are quite an effort, this number of checks is still computationally feasible.

Thus the application of a JPEG2000 Part 2 encoder and KDWP can not be considered secure (neither MR-secure nor MQ-secure) due to the restrictions in terms of admissible WP. For non-standard WP security is expected to be increased.

5.3.2 Non-standard WP

The complexity an adversary has to face is given by the entropy of the distribution on the WP on a certain resolution. Tables 7 and 8 summarize the results, while details on the computation are given in Appendix 2). An attacker has to try $2^{H(X)-1}$ WP on average to decode a certain resolution/quality.

In the isotropic case the compression oriented distribution all obtained entropy values are below state-of-the-art cipher key-length (AES has at least 128 bit). The uniform distribution also results in entropy values below AES minimum key length, except for the highest resolution. The anisotropic case with uniform distribution shows higher entropy values, but at a quality of 1/16 security drops far below AES minimum key length. The anisotropic case with the compression oriented

Table 7 Entropy for the distribution on resolutions of the isotropic compression-oriented distribution (*g* = 5, *b* = 1/4, *c* = 0) and the uniform distribution

Res. at level	Quality	Entropy (iso)	Entropy (isouni)
0	1	114.0	261.6
1	1/4	28.7	65.4
2	1/16	7.5	16.3
3	1/64	2.2	4.1
4	1/256	0.8	1.0
5	1/1024	0.0	0.0

Table 8 Entropy for the distribution of decomposition structures of a resolution as induced by the anisotropic uniform distribution (*g* = 10, *b* = 1/4, *c* = 0)

Res. at level	Quality	Entropy (anisouni)
0	1	1321.9
1	1/2	660.5
2	1/4	329.7
3	1/8	164.4
4	1/16	81.7
5	1/32	40.3
6	1/64	19.7
7	1/128	9.3
8	1/256	4.2
9	1/512	1.5
10	1/1024	0.0

distribution is expected to be far below the security levels of the uniform distribution. In general an attacker can compute lower resolutions, while only the higher resolutions remain well-protected.

6. Discussion

Media encryption schemes relying on secret transform domains have been proposed mainly motivated by the significant reduction of the computational demand for encryption as compared to traditional transparent encryption methods and by potential capabilities in encrypted domain signal processing. A thorough analysis of the properties of the KDWP approach has revealed that significant disadvantages as compared to conventional compression and encryption schemes exist.

- *Compression impact*: The KDWP approach obviously reduces compression performance. If all possible subband structures are equally likely, the approach is not be suitable for application due to the high variance of obtained compression results. The approach of pruning the set of all subband structures (compression oriented distribution) improves the compression performance, however, still with this technique, a significant loss exists. On the other hand, conventional encryption of course does not at all influence compression performance.
- *Computational demand*: Although the additional complexity of encryption with KDWP in a compression framework, such as JPEG2000, seems to be negligible at a first glance, our careful analysis and our evaluation results clearly show that runtime advantages can not be achieved compared to state-of-the-art cryptographic ciphers (AES). In general we advise to carefully reconsider statements about an improved runtime performance of transform based image and video encryption schemes.
- *Security*: From a security point of view, state-of-the-art cryptographic ciphers are superior to KDWP. The KDWP schemes cannot prevent access to lower resolutions and are thus less secure in terms MQ-security. Within the standardized framework of JPEG2000 Part 2, the KDWP approach is completely insecure, i.e., an attacker is able to reconstruct the entire image.

From the proposed KDWP schemes, the anisotropic compression oriented scheme is best performing. However, even this approach is overall outperformed by conventional compression and encryption, which shows no decrease in compression performance, better runtime performance and cryptographic security.

Apart from an improved runtime performance, image encryption schemes can offer an improved functionality. On the one hand improved functionality could be suitability for specific applications, such as transparent encryption. However, transparent encryption can also be implemented in the conventional approach [10], the small resolution portion of the JPEG2000 file is simply left in plaintext. A standardized tool, namely JPSEC, can be employed to signal all the meta-data (keys, plaintext parts). Even completely JPEG2000-compliant implementations are possible [10], which do not require special software for decoding, a great advantage for real-world deployment of transparent encryption. Transparent encryption with KDWP requires a special decoder, i.e., an attacker's decoder, that can cope with wrong or missing WP structure information, while JPSEC-based approaches require a JPSEC-capable decoder. Thus, the support of transparent encryption by KDWP is not an advantage compared to other approaches, which are more flexible. The conventional approach offers full confidentiality/cryptographic security (MP-security).

Image encryption schemes may also allow special encrypted domain processing, which may justify their application although there are disadvantages in compression, computational demand and security. A good example is encrypted content that can still be robustly watermarked; this feature would outweigh many disadvantages.

6..1 Encrypted domain processing

In fact for KDWP in JPEG2000, the encrypted domain is a scrambled JPEG2000 bitstream and only signal processing operations that can be conducted in this domain are possible. In fact the only possible processing is the truncation of the scrambled bitstream, which results in an almost rate-distortion optimal rate adaptation (in case the underlying bitstream is in quality progression order). However, truncation of an encrypted bitstream is also possible with conventional block ciphers if the are used in the proper mode, e.g., counter mode.

Signal processing operations which rely on transform coefficient data cannot be applied to KDWP protected data as these data can not be decoded. Given the results of our evaluation and analysis (decreased compression performance, increased computational complexity, decreased security, no other advantages) it has to be stated that hardly any sensible realistic application scenarios can be identified for KDWP-based encryption at the present time.

7. Conclusion

A primary argument for proposing KDWP-based encryption has always been its "lightweight" nature, introduced by shifting complexity from encryption into the compression pipeline. When comparing JPEG2000 encryption with key-dependent wavelet packets (KDWP) to the conventional approach (AES encryption), we have assessed an overall increase in computational complexity through additional complexity introduced in the compression step.

Signal processing in the encrypted domain, often used as a second argument favoring transform-based image encryption schemes, does not offer advantages compared to the appropriate application of conventional ciphers. The security of JPEG2000 encryption with KDWP has been analyzed in depth and has been found to be less secure than conventional encryption, as smaller resolution images remain accessible. Security can not be achieved at all if only permissible subband structures of JPEG2000 Part 2 are employed.

All these facts taken together with a slight decrease in compression efficiency as compared to classical (pyramidal) JPEG2000 make KDWP-based encryption approaches not suitable for most application scenarios.

The presented assessment should serve as guideline in the future development of image and video encryption schemes. The following general conclusions may be drawn: Computational complexity will need to be carefully (re)considered for transform-based image and video encryption schemes. Security has to be analyzed with respect to a multimedia specific security notion (MQ-security). Image and video encryption schemes have to provide conclusive evidence for improved functionality, such as encrypted domain signal processing.

Appendix 1: Details on the entropy computation of WP distributions

A WP (wavelet packet subband structure) is derived by the following randomized algorithm (see Section 2..2.2): every subband at depth l is further decomposed with a certain probability p_l, which may only depend on the depth l.

Game tree

This randomized algorithm can be illustrated with the corresponding *game tree.* A game tree $\mathcal{G}_g$ is the tuple (V, E, l, p):

$$V \ldots \text{set of vertices}, \quad E \subset V \times V \ldots \text{set of edges}, \quad l : E \to \{0,1\}^*, \quad p : E \to \mathbb{R}^+$$

The vertices of a game tree correspond to a certain WP. The edge label (l) in a game tree indicates decomposition decisions and is associated with the probability p that this decomposition decisions are selected in the randomized algorithm. In Figure 8a game tree is illustrated, showing the edge labels and the vertices. The first decision is whether the entire image gets decomposed (split into four subbands), there are two outcomes, an edge labeled with a "0" indicates no decomposition, an edge labeled with a "1" indicates a decomposition into four distinct subbands. A "0" in the label of a edge is replaced by "0000" in the label of a next edge, in order to obtain same length labels at a certain depth of the tree. A "1" in the label of a edge is replaced by one of the strings "0000",... ,"1111" in the label of the next edge, which indicates the further decompositions of the 4 subbands. I.e., the string "0000" indicates that no subband is decomposed, the string "0001" indicates that the last subband (HH) is decomposed, ..., and the string "1111" indicates that all subbands are decomposed. A label uniquely identifies a further decomposition. Note that 0^{16} denotes the string 0000000000000000 and analogous is the meaning of 1^{16}. A unique code for a node is derived by a separated concatenation of the edge labels from a path from the root to the node. We use "," as a separator. Each node corresponds to a certain WP. Figure 8b shows how the edges and edge labels are determined by the predecessor edge. The function pre: $E \to E$ gives the predecessor edge of an edge, e.g., in Figure 8b the predecessor of the edge labeled y, $l(e) = y$, is the edge labeled x, i.e., $l(\text{pre}(e)) = x$. $S(x, r_1, \ldots, r_n)$ denotes the string obtained by applying the substitution rules r_1 to r_n to string x. If there is a complex rule, which allow choices, i.e., the right side of one rule is a set of strings, $S(x, r_1,\ldots,r_n)$ denotes the set of all possible substitutions. In Figure 8b new edges are added to the vertex (s, x) depending on the edge label x: the label of the first edge is obtained by substituting every "0" in x by "0000" and every "1" by "0000". The labels of all outgoing edges of (s, x) are obtained by all possible substitutions of a "1" in x. The last edge (see Figure 8b) is obtained by substituting all "1"s by "1111". A predecessor label determines the edges and edge labels in the following way:

$$l(e) = y, \quad l(\text{pre}(e)) = x, \quad x \in \{0,1\}^n, \quad x = (x_1, \ldots, x_n)\,, \quad x_i \in \{0,1\}$$
$$y \in S\,(x, 0 \to 0000, 1 \to \{0000, \ldots, 1111\}) \subset \{0,1\}^{4n}$$

The function p assigns each edge a probability (the probabilities of the outgoing edges of a node sum up to 1). The probability of an edge can be determined by its label y and the label of its predecessor x, by simply considering the number of actual decomposition decisions (Σy_i) and the number of maximally possible decomposition decisions ($4\Sigma x_i$):

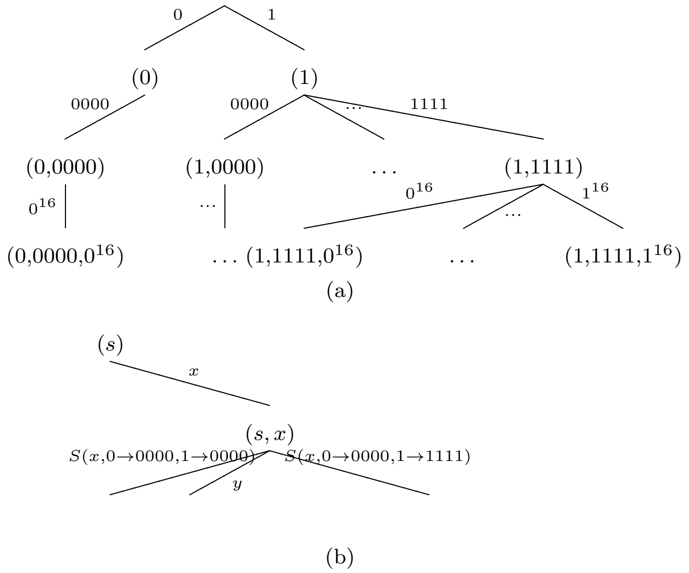


Figure 8 Random generation of isotropic WP, the game tree.

$$l(e) = y, \quad l(\mathrm{pre}(e)) = x, \quad p(y) = p_l^{\Sigma y_i}(1 - p_l)^{4\Sigma x_i - \Sigma yi}$$

Every leaf of a game tree with depth *g* corresponds to exactly one WP ψ with maximum decomposition depth *g*, the probability of a WP ψ is derived by the product of the edge weights *p* of the path from the root to the leaf.

$$\psi \in V : p(\psi) = \Pi_{e\in \mathrm{Path(root},\psi)}p(e)$$

We denote the entropy of corresponding distribution for a game tree $\mathcal{G}_g$ by:

$$H(\mathcal{G}_g) = \sum_{\psi\in \mathrm{Leaves}(\mathcal{G}_g)} -p(\psi)1\mathrm{d}p(\psi)$$

However, as the number of leaves at depth *g* is *Q*(*g*) the computation of the entropy of the distribution on WPs on the basis of this formula is soon infeasible with growing *g*.

Cumulative game tree

A simpler representation of a game tree $\mathcal{G}_g$ is its corresponding *cumulative game tree* (CuGa-Tree), $\mathcal{C}_g$. A CuGa-Tree $\mathcal{C}_g$ is the tuple (*V, E, l, p, n*):

$$l : E \rightarrow \mathbb{N}, \quad p : E \rightarrow \mathbb{R}^+, \quad n : E \rightarrow \mathbb{N}$$

A CuGa-Tree summarizes the edges of a node with the with the same probability *p*, i.e., with the same number of decomposition decisions, i.e., with the same number of "1"s in the edge label of the game tree. Thus the edge label of a CuGa-Tree indicates the number of decomposition decisions, i.e., the number of subbands which are further decomposed. We have to keep track how many edges of the game tree are summarized by an edge of a CuGa-Tree, therefore we introduce a weight function $n : E \rightarrow \mathbb{N}$. A CuGa-Tree with depth 2 is shown in Figure 9.

The edges and edge labels are determined by the predecessor edge (see Figure 9b): the successors of an edge with label *l*(*e*) = *i* (this number of subbands have been decomposed) can be in the range of 0 to 4*i*, as every subband may have up to four children:

$$l(\mathrm{pre}(e)) = i, \quad l(e) \in \{0, \ldots, 4i\}$$

The probability for an edge is similar to game trees:

$$p(e) = p_l^{l(e)}(1 - p_l)^{4l(\mathrm{pre}(e))-l(e)}$$

The number of edges in the game tree with the same probability is derived by counting the number of edges with the same number of "1"s, i.e., decomposition decisions in the edge label: There are 4*l*(pre(*e*)) possible positions for *l*(*e*) "1"s and thus there are $\binom{4l\,(P\mathrm{pre}(e))}{l(e)}$ edges with the same probability in the game tree:

$$n(e) = \binom{4l\,(P\mathrm{pre}(e))}{l(e)}$$

The probability *p* and weight *n* are defined for vertices $\psi \in V$ in the following way:

$$p(\psi) = \Pi_{e\in \mathrm{Path(root},\psi)}p(e)$$
$$n(\psi) = \Pi_{e\in \mathrm{Path(root},\psi)}n(e)$$

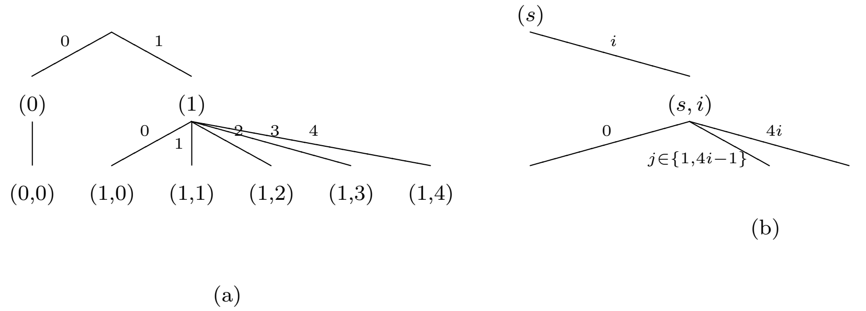


Figure 9 Recursive definition of CuGa-Trees.

The entropy of the corresponding distribution of a CuGa-Tree $\mathcal{C}_g$ can be computed by:

$$H\left(\mathcal{C}_g\right) = \sum_{\psi \in \text{Leaves}(\mathcal{C}_g)} -n(\psi)p(\psi)1\text{d}p(\psi)$$

The nodes can be uniquely identified by the path from the root, i.e., by tuple of edge labels. A node at depth g is a g-tuple of edge labels $(x_1,..., x_g)$. The set of all nodes at depth g is given by $\{(x_1,...,x_g)|x_1 \in \{0,1\}, x_{i+1} \leq 4x_i\}$.

Appendix 2: Details on the entropy computation of distributions of decomposition structures on resolutions

In order to assess the MQ-security of KDWP we need to compute the entropy of the resulting distribution of the decomposition structures on a resolution, i.e., on the subband is the result of always decomposing the low pass band further (no high pass filtering, i.e., either the LL, LX or XL subband). Thus only the case of a low pass band decomposition is of interest up to the depth d of the targeted resolution (see Figure 10). The entropy of the decomposition structures of a resolution corresponds to the entropy of the tree of Figure 10, the depth of the resolution has to be considered for the split-probability in sub-tree $\mathcal{C}_{g-d}$, which is indicated by the notation $\mathcal{C}_{g-d}(p_d)$. Thus entropy computation is straight-forward, namely the entropy of the decomposition structures for a resolution d can be computed by:

$$q = 1 - \Pi_{l=0}^{d-1} p_l, \quad p = \Pi_{l=0}^{d-1} p_l$$
$$H\left(\mathcal{R}_{g,d}\right) = q1\text{dl}/q + p1\text{dl}/p + pH\left(\mathcal{C}_{g-d}\left(p_d\right)\right)$$

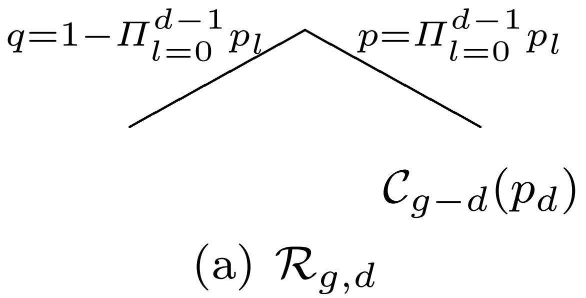


Figure 10 The entropy of distributions of decomposition structures on resolutions.

Endnotes

[a]http://jj2000.epfl.ch/. [b]Linux binaries in version 6.3.1 from http://www.kakadusoftware.com.

Acknowledgements
This work was funded by the Austrian Science Fund under Project 19159. The authors thank Stefan Huber and Rade Kutil, and Roland Kwitt for the valuable discussions and contributions.

Author details
[1]University of Applied Sciences, Urstein Sued 1, 5412 Puch/Salzburg Austria
[2]Dept. of Computer Sciences, University of Salzburg, Jakob Haringer Str. 2, 5020 Salzburg, Austria

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

Received: 3 June 2011 Accepted: 13 April 2012 Published: 13 April 2012

References
1. BM Macq, JJ Quisquater, Cryptology for digital TV broadcasting. Proc IEEE. **83**(6):944–957 (1995)
2. ISO/IEC 15444-1, Information technology–JPEG2000 image coding system. Part 1: Core coding system. (2000)
3. D Taubman, M Marcellin, *JPEG2000–Image Compression Fundamentals, Standards and Practice*. (Kluwer Academic Publishers, Boston, 2002)
4. Digital Cinema Initiatives, LLC (DCI), Digital cinema system specification v1.1, online presentation. (2007)
5. R Grosbois, P Gerbelot, T Ebrahimi, Authentication and access control in the JPEG2000 compressed domain. in *Applications of Digital Image Processing XXIV, ser Proceedings of SPIE*, vol. 4472, ed. by Tescher A (San Diego, CA, USA, 2001), pp. 95–104
6. H Kiya, D Imaizumi, O Watanabe, Partial-scrambling of image encoded using JPEG2000 without generating marker codes. in Proceedings of the IEEE International Conference on Image Processing (ICIP'03), vol. III. (Barcelona, Spain, 2003), pp. 205–208
7. T Stütz, A Uhl, On format-compliant iterative encryption of JPEG2000. *Proceedings of the Eighth IEEE International Symposium on Multimedia (ISM'06)*. (IEEE Computer Society, San Diego, CA, USA, 2006), pp. 985–990
8. M Grangetto, E Magli, G Olmo, Multimedia selective encryption by means of randomized arithmetic coding. IEEE Trans Multimedia. **8**(5):905–917 (2006)

9. Y Yang, BB Zhu, Y Yang, S Li, N Yu, Efficient and syntax-compliant JPEG2000 encryption preserving original fine granularity of scalability. EURASIP J Inf Security (2007). 2007:056365
10. T Stütz, A Uhl, On efficient transparent JPEG2000 encryption. *Proceedings of ACM Multimedia and Security Workshop, MM-SEC '07*. (New York, NY, USA, ACM, 2007), pp. 97–108
11. D Engel, T Stütz, A Uhl, A survey on JPEG2000 encryption. Multimedia Syst. **15**(4):243–270 (2009)
12. ISO/IEC 15444-8, Information technology–JPEG2000 image coding system. Part 8: Secure JPEG2000. (2007)
13. ITU-T T.807, Information technology–JPEG2000 image coding system. Part 8: Secure JPEG2000. (2006)
14. B Macq, J Quisquater, Digital images multiresolution encryption. J Interactive Multimedia Association Intellectual Property Project. **1**(1):179–206 (1994)
15. T Stütz, V Pankajakshan, F Autrusseau, A Uhl, H Hofbauer, Subjective and objective quality assessment of transparently encrypted JPEG2000 images. *Proceedings of the ACM Multimedia and Security Workshop (MMSEC'10)*. (ACM, Rome, Italy, 2010), pp. 247–252
16. D Engel, A Uhl, Secret wavelet packet decompositions for JPEG2000 lightweight encryption. in Proceedings of 31st International Conference on Acoustics, Speech, and Signal Processing, ICASSP'06, vol. V. (Toulouse, France, 2006), pp. 465–468
17. G Unnikrishnan, K Singh, Double random fractional fourier-domain encoding for optical security. Opt Eng. **39**(11):2853–2859 (2000)
18. L Vorwerk, T Engel, C Meinel, A proposal for a combination of compression and encryption. in ser Proceedings of SPIE, vol. 4067. (Perth, Australia, 2000), pp. 694–702
19. A Pommer, A Uhl, Wavelet packet methods for multimedia compression and encryption. *Proceedings of the 2001 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*. (Victoria, Canada: IEEE Signal Processing Society, 2001), pp. 1–4
20. A Pommer, A Uhl, Lightweight protection of visual data using high-dimensional wavelet parametriza-tion. in *Image Analysis and Processing - ICIAP 2005, ser. Lecture Notes on Computer Science*, vol. 3617, ed. by Roli F, Vitulano S (Cagliari, Italy, 2005), pp. 645–652
21. A Uhl, A Pommer, Are parameterised biorthogonal wavelet filters suited (better) for selective encryption? in *Multimedia and Security Workshop 2004*, ed. by Dittmann J, Fridrich J (Magdeburg, Germany, 2004), pp. 100–106
22. A Pommer, A Uhl, Selective encryption of wavelet packet subband structures for secure transmission of visual data. in *Multimedia and Security Workshop, ACM Multimedia*, ed. by Dittmann J, Fridrich J, Wohlmacher P (Juan-les-Pins, France, 2002), pp. 67–70
23. A Pommer, A Uhl, Selective encryption of wavelet-packet encoded image data–efficiency and security. ACM Multimedia Syst (Special issue on Multimedia Security). **9**(3):279–287 (2003)
24. T Köckerbauer, M Kumar, A Uhl, Lightweight JPEG2000 confidentiality for mobile environments. in Proceedings of the IEEE International Conference on Multimedia and Expo, ICME'04, vol. 2. (Taipei, Taiwan, 2004), pp. 1495–1498
25. D Engel, A Uhl, Parameterized biorthogonal wavelet lifting for lightweight JPEG2000 transparent encryption. *Proceedings of ACM Multimedia and Security Workshop, MM-SEC'05*. (New York, NY, USA, 2005), pp. 63–70
26. D Engel, R Kutil, A Uhl, A symbolic transform attack on lightweight encryption based on wavelet filter parameterization. *Proceedings of ACM Multimedia and Security Workshop, MM-SEC'06*. (Geneva, Switzerland, 2006), pp. 202–207
27. W Zeng, S Lei, Efficient frequency domain selective scrambling of digital video. **5**(1):118–129 (2003)
28. S-KA Yeung, S Zhu, B Zeng, Partial video encryption based on alternating transforms. IEEE Signal Process Lett. **16**(10):893–896 (2009)
29. ISO/IEC 15444-2, Information technology–JPEG2000 image coding system. Part 2: Extensions. (2004)
30. D Engel, A Uhl, Lightweight JPEG2000 encryption with anisotropic wavelet packets. *Proceedings of International Conference on Multimedia & Expo, ICME'06*. (Toronto, Canada, 2006), pp. 2177–2180
31. D Engel, A Uhl, An evaluation of lightweight JPEG2000 encryption with anisotropic wavelet packets. in *Security, Steganography, and Watermarking of Multimedia Contents IX, ser Proceedings of SPIE*, ed. by Delp EJ, Wong PW (SPIE, San Jose, CA, USA, 2007), pp. 65 051S1–65 051S10
32. M Wickerhauser, *Adapted Wavelet Analysis from Theory to Software*. (A.K. Peters, Wellesley, Mass, 1994)
33. K Ramchandran, M Vetterli, Best wavelet packet bases in a rate-distortion sense. IEEE Trans Image Process. **2**(2):160–175 (1993)
34. T Stütz, B Mühlbacher, A Uhl, Best wavelet packet bases in a JPEG2000 rate-distortion sense: the impact of header data. *Proceedings of the IEEE International Conference on Multimedia & Expo, ICME'10*. (Singapore, 2010), pp. 19–24
35. T Stütz, A Uhl, Efficient wavelet packet basis selection in JPEG2000. *Proceedings of the IEEE International Conference on Image Processing, ICIP'11*. (Brussels, Belgium, 2011), pp. 317–320
36. T Stütz, A Uhl, Efficient and rate-distortion optimal wavelet packet basis selection in JPEG2000. IEEE Trans Multimedia. (in press)
37. R Kutil, D Engel, Methods for the anisotropic wavelet packet transform. Appl Comput Harmonic Anal. **25**(3):295–314 (2008)
38. A Pommer, A Uhl, Selective encryption of wavelet packet subband structures for obscured transmission of visual data. *Proceedings of the 3rd IEEE Benelux Signal Processing Symposium (SPS 2002)*. (IEEE Benelux Signal Processing Chapter, Leuven, Belgium, 2002), pp. 25–28
39. CE Shannon, Communication theory of secrecy systems. Bell Syst Tech J. **28**, 656–715 (1949)
40. O Goldreich, *The Foundations of Cryptography*. (Cambridge University Press, Cambridge, 2001)
41. M Bellare, T Ristenpart, P Rogaway, T Stegers, Format-preserving encryption. in Proceedings of Selected Areas in Cryptography, SAC'09, vol. 5867. (Calgary, Canada, 2009), pp. 295–312
42. T Stütz, A Uhl, Efficient format-compliant encryption of regular languages: block-based cycle-walking. in *Proceedings of the 11th Joint IFIP TC6 and TC11 Conference on Communications and Multimedia Security, CMS'10, ser. IFIP Advances in Information and Communication Technology*, vol. 6109, ed. by Decker BD, Schaum?ü?ller-Bichl I (Springer, Linz, Austria, 2010), pp. 81–92
43. A Said, Measuring the strength of partial encryption schemes. Proceedings of the IEEE International Conference on Image Processing (ICIP'05). **2**, 1126–1129 (2005)
44. Y Mao, M Wu, A joint signal processing and cryptographic approach to multimedia encryption. IEEE Trans Image Process. **15**(7):2061–2075 (2006)

doi:10.1186/1687-417X-2012-2
Cite this article as: Engel *et al.*: **Assessing JPEG2000 encryption with key-dependent wavelet packets.** *EURASIP Journal on Information Security* 2012 **2012**:2.