

## JPEG 2000 still image coding versus other standards

D. Santa-Cruz<sup>a</sup>, T. Ebrahimi<sup>a</sup>, J. Askelöf<sup>b</sup>, M. Larsson<sup>b</sup> and C. A. Christopoulos<sup>b</sup>

<sup>a</sup>Signal Processing Laboratory – Swiss Federal Institute of Technology  
CH-1015 Lausanne, Switzerland

E-mail: {Diego.SantaCruz, Touradj.Ebrahimi}@epfl.ch

<sup>b</sup>Ericsson Research, Corporate Unit  
S-164 Stockholm, Sweden

E-mail: {Joel.Askelof, Mathias.Larsson, Charilaos.Christopoulos}@era.ericsson.se

### ABSTRACT

JPEG 2000, the new ISO/ITU-T standard for still image coding, is about to be finished. Other new standards have been recently introduced, namely JPEG-LS and MPEG-4 VTC. This paper compares the set of features offered by JPEG 2000, and how well they are fulfilled, versus JPEG-LS and MPEG-4 VTC, as well as the older but widely used JPEG and more recent PNG. The study concentrates on compression efficiency and functionality set, while addressing other aspects such as complexity. Lossless compression efficiency as well as the fixed and progressive lossy rate-distortion behaviors are evaluated. Robustness to transmission errors, Region of Interest coding and complexity are also discussed. The principles behind each algorithm are briefly described. The results show that the choice of the “best” standard depends strongly on the application at hand, but that JPEG 2000 supports the widest set of features among the evaluated standards, while providing superior rate-distortion performance in most cases.

**Keywords:** image coding, standards, wavelets, DWT, DCT, JPEG, JPEG-LS, JPEG 2000, MPEG-4, PNG

### 1. INTRODUCTION

It has been three years since the call for proposals<sup>1</sup> for the next ISO/ITU-T standard for compression of still images, JPEG 2000, has been issued. Now JPEG 2000 Part I (the core system) is in its final stage to become an International Standard (IS). It has been promoted to Final Committee Draft (FCD)<sup>2</sup> in March 2000 and will reach IS status by the end of the same year. A great effort has been made to deliver a new standard for today's and tomorrow's applications, by providing features inexistent in previous standards, but also by providing higher efficiency for features that exist in others. Now that the new standard is nearing finalization, a trivial question would be: what are the features offered by JPEG 2000 but also how well are they fulfilled when compared to other standards offering the same features. This paper aims at providing an answer to this simple but somewhat complex question. Section 2 provides a brief overview of the techniques compared, with special attention on new features of JPEG 2000 such as Region of Interest (ROI) coding. Section 3 explains the comparison methodology employed in the results shown in section 4 and conclusions are drawn in section 5.

### 2. OVERVIEW OF STILL IMAGE CODING STANDARDS

For the purpose of this study we compare the coding algorithm in JPEG 2000 standard to the following three standards: JPEG<sup>3</sup>, MPEG-4 Visual Texture Coding (VTC)<sup>4</sup> and JPEG-LS<sup>5</sup>. In addition, we also include PNG<sup>6</sup>. The reasons behind this choice are as follows. JPEG is one of the most popular coding techniques in imaging applications ranging from Internet to digital photography. Both MPEG-4 VTC and JPEG-LS are very recent standards that start appearing in various applications. It is only logical to compare the set of features offered by JPEG 2000 standard not only to those offered in a popular but older standard (JPEG), but also to those offered in most recent ones using newer state-of-the-art technologies. Although PNG is not formally a standard and is not based on state-of-the-art techniques, it is becoming increasingly popular for Internet based applications. PNG is also undergoing standardization by ISO/IEC JTC1/SC24 and will eventually become ISO/IEC international standard 15948.

## 2.1. JPEG

This is the very well known ISO/ITU-T standard created in the late 1980s. There are several modes defined for JPEG<sup>3</sup>, including baseline, lossless, progressive and hierarchical. The baseline mode is the most popular one and supports lossy coding only. The lossless mode is not popular but provides for lossless coding, although it does not support lossy.

In the baseline mode, the image is divided in 8x8 blocks and each of these is transformed with the DCT. The transformed blocks are quantized with a uniform scalar quantizer, zig-zag scanned and entropy coded with Huffman coding. The quantization step size for each of the 64 DCT coefficients is specified in a quantization table, which remains the same for all blocks. The DC coefficients of all blocks are coded separately, using a predictive scheme. Hereafter we refer to this mode simply as JPEG.

The lossless mode is based on a completely different algorithm, which uses a predictive scheme. The prediction is based on the nearest three causal neighbors and seven different predictors are defined (the same one is used for all samples). The prediction error is entropy coded with Huffman coding. Hereafter we refer to this mode as L-JPEG.

The progressive and hierarchical modes of JPEG are both lossy and differ only in the way the DCT coefficients are coded or computed, respectively, when compared to the baseline mode. They allow a reconstruction of a lower quality or lower resolution version of the image, respectively, by partial decoding of the compressed bitstream. Progressive mode encodes the quantized coefficients by a mixture of spectral selection and successive approximation, while hierarchical mode uses a pyramidal approach to computing the DCT coefficients in a multi-resolution way.

## 2.2. MPEG-4 VTC

MPEG-4 Visual Texture Coding (VTC) is the algorithm used in MPEG-4<sup>4</sup> to compress visual textures and still images, which are then used in photo realistic 3D models, animated meshes, etc., or as simple still images. It is based on the discrete wavelet transform (DWT), scalar quantization, zero-tree coding and arithmetic coding. The DWT is dyadic and uses a Daubechies (9,3) tap biorthogonal filter. The quantization is scalar and can be of three types: single (SQ), multiple (MQ) and bi-level (BQ). With SQ each wavelet coefficient is quantized once, the produced bitstream not being SNR scalable. With MQ a coarse quantizer is used and this information coded. A finer quantizer is then applied to the resulting quantization error and the new information coded. This process can be repeated several times, resulting in limited SNR scalability. BQ is essentially like SQ, but the information is sent by bitplanes, providing general SNR scalability. Two scanning modes are available: tree-depth (TD), the standard zero-tree scanning, and band-by-band (BB). Only the latter provides for resolution scalability. The produced bitstream is resolution scalable at first, if BB scanning is used, and then SNR scalable within each resolution level, if MQ or BQ is used.

A unique feature of MPEG-4 VTC is the capability to code arbitrarily shaped objects. This is accomplished by the means of a shape adaptive DWT and MPEG-4's shape coding. Several objects can be encoded separately, possibly at different qualities, and then composited at the decoder to obtain the final decoded image. On the other hand, MPEG-4 VTC does not support lossless coding.

## 2.3. JPEG-LS

JPEG-LS<sup>5</sup> is the latest ISO/ITU-T standard for lossless coding of still images. It also provides for "near-lossless" compression. Part-I, the baseline system, is based on adaptive prediction, context modeling and Golomb coding. In addition, it features a flat region detector to encode these in run-lengths. Near-lossless compression is achieved by allowing a fixed maximum sample error. Part-II will introduce extensions such as an arithmetic coder, but is still under preparation. This algorithm was designed for low-complexity while providing high lossless compression ratios. However, it does not provide support for scalability, error resilience or any such functionality.

## 2.4. PNG

Portable Network Graphics (PNG)<sup>6</sup> is a W3C recommendation for coding of still images which has been elaborated as a patent free replacement for GIF, while incorporating more features than this last one. It is based on a predictive scheme and entropy coding. The prediction is done on the three nearest causal neighbors and there are five predictors that can be selected on a line-by-line basis. The entropy coding uses the Deflate algorithm of the popular Zip file compression utility,

which is based on LZ77 coupled with Huffman coding. PNG is capable of lossless compression only and supports gray scale, paletted color and true color, an optional alpha plane, interlacing and other features.

## 2.5. JPEG 2000

JPEG 2000<sup>2</sup>, as noted previously, is the next ISO/ITU-T standard for still image coding. In the following, we restrict the description to Part I of the standard, which defines the core system. Part II will provide various extensions for specific applications, but is still in preparation. JPEG 2000 is based on the discrete wavelet transform (DWT), scalar quantization, context modeling, arithmetic coding and post-compression rate allocation. The DWT is dyadic and can be performed with either the reversible Le Gall (5,3) taps filter<sup>9</sup>, which provides for lossless coding, or the non-reversible Daubechies (9,7) taps biorthogonal one<sup>10</sup>, which provides for higher compression but does not do lossless. The quantizer follows an embedded dead-zone scalar approach and is independent for each sub-band. Each sub-band is divided into rectangular blocks (called code-blocks in JPEG 2000), typically 64x64, and entropy coded using context modeling and bit-plane arithmetic coding. The coded data is organized in so called *layers*, which are quality levels, using the post-compression rate allocation and output to the code-stream in packets. The generated code-stream is parseable and can be resolution, layer (i.e. SNR), position or component progressive, or any combination thereof. JPEG 2000 also supports a number of functionalities, many of which are inherent from the algorithm itself. Examples of this is random access, which is possible because of the independent coding of the code-blocks and the packetized structure of the codestream. Another such functionality is the possibility to encode images with arbitrarily shaped Regions of Interest (ROI)<sup>11</sup>. The fact that the subbands are encoded bitplane by bitplane makes it possible to select regions of the image that will precede the rest of the image in the codestream. By scaling the sub-band samples so that the bitplanes encoded first only contain ROI information and following bitplanes only contain background information. The only thing the decoder needs to receive is the factor by which the samples were scaled. The decoder can then invert the scaling based only on the amplitude of the samples. Other supported functionalities are error-resilience, random access, multicomponent images, palletized color, compressed domain lossless flipping and simple rotation, to mention a few.

## 3. COMPARISON METHODOLOGY

Although one of the major, and often only, concerns in coding techniques has been that of compression efficiency, it is not the only factor that determines the choice of a particular algorithm for an application. Most applications also require other features in a coding algorithm than simple compression efficiency. This is often referred to as functionalities. Examples of such functionalities are ability to distribute quality in a non-uniform fashion across the image (e.g., ROI), or resiliency to residual transmission errors that occur in mobile channels. In this paper we report on compression efficiency, since it is still one of the top priorities in many imaging products, but we also devote attention to complexity and functionalities. In the next section we summarize the results of the study as long as the considered functionalities are concerned.

### 3.1. Compression efficiency

Compression efficiency is measured for lossless and lossy compression. For lossless coding it is simply measured by the achieved compression ratio for each one of the test images. For lossy coding the root mean square error (RMSE) is used, as well as the corresponding peak signal to noise ratio (PSNR), defined as

$$-20 \log_{10} \frac{RMSE}{2^{b-1}}$$

where  $b$  is the bit depth of the original image.

Although RMSE and PSNR are known to not always faithfully represent visual quality, it is the only established, well-known, objective measure that works reasonably well across a wide range of compression ratios.

For images encoded with a Region of Interest (ROI) the RMSE, as well as the corresponding PSNR, are calculated both for the ROI and for the entire image.

### 3.2. Complexity

Evaluating complexity is a difficult issue, with no well-defined measure. It means different things for different applications. It can be memory bandwidth, total working memory, number of CPU cycles, number of hardware gates, etc. Furthermore,

these numbers are very dependent on the optimization, targeted applications and other factors of the different implementations.

As a rough indication of complexity we provide the run times of the different algorithms on a Linux based PC. This only gives an appreciation of the involved complexity.

### 3.3. Functionalities

Comparing how well different functionalities are fulfilled in the different standards is also a difficult issue. In the next section we provide a functionality matrix that indicates the set of supported features in each standard and an appreciation of how well they are fulfilled. Although in most cases this appreciation is based on the other results presented here, in some others it is based on the capabilities provided by the different algorithms.

## 4. RESULTS

The algorithms have been evaluated with seven images from the JPEG 2000 test set, covering various types of imagery. The images “bike” (2048x2560) and “cafe” (2048x2560) are natural, “cmpnd1” (512x768) and “chart” (1688x2347) are compound documents consisting of text, photographs and computer graphics, “aerial2” (2048x2048) is an aerial photography, “target” (512x512) is a computer generated image and “us” (512x448) an ultra scan. All these images have a depth of 8 bits per pixel.

The results have been generated on a PC with a 550 MHz Pentium™ III processor, 512 kB of cache and 512 MB of RAM under Linux 2.2.12. The software implementations used for coding the images are: the JPEG 2000 Verification Model (VM) 6.1 (ISO/IEC JTC1/SC29/WG1 N 1580), the MPEG-4 MoMuSys VM of Aug. 1999 (ISO/IEC JTC1/SC29/WG11 N 2805), the Independent JPEG Group JPEG implementation (<http://www.ijg.org/>), version 6b, the SPMG JPEG-LS implementation of the University of British Columbia (<http://spmge.ece.ubc.ca/>), version 2.2, the Lossless JPEG codec of Cornell University (<ftp://ftp.cs.cornell.edu/pub/multimed>), version 1.0, and the libpng implementation of PNG (<ftp://ftp.uu.net/graphics/png>), version 1.0.3.

### 4.1. Lossless compression

Table 1 summarizes the lossless compression efficiency of lossless JPEG (L-JPEG), JPEG-LS, PNG and JPEG 2000 for all the test images. For JPEG 2000 the reversible DWT filter, referred to as JPEG 2000<sub>R</sub>, has been used. In the case of L-JPEG optimized Huffman tables and the predictor yielding the best compression performance have been used for each image. For PNG the maximum compression setting has been used, while for JPEG-LS the default options were chosen. MPEG-4 VTC is not considered, as it does not provide a lossless functionality.

**Table 1.** Lossless compression ratios

	JPEG 2000 <sub>R</sub>	JPEG-LS	L-JPEG	PNG
bike	1.77	1.84	1.61	1.66
cafe	1.49	1.57	1.36	1.44
cmpnd1	3.77	6.44	3.23	6.02
chart	2.60	2.82	2.00	2.41
aerial2	1.47	1.51	1.43	1.48
target	3.76	3.66	2.59	8.70
us	2.63	3.04	2.41	2.94
<b>average</b>	<b>2.50</b>	<b>2.98</b>	<b>2.09</b>	<b>3.52</b>

It can be seen that in almost all cases the best performance is obtained by JPEG-LS. JPEG 2000 provides, in most cases, competitive compression ratios with the added benefit of scalability. PNG performance is similar to the one of JPEG 2000. As for lossless JPEG, it does not perform as well as the other, more recent, standards. One notable exception to the general trend is the “target” image, which contains mostly patches of constant gray level as well as gradients. For this type of images, PNG provides the best results, probably because of the use of LZ77. Another exception is the “cmpnd1” image, in which JPEG-LS and PNG achieve much larger compression ratios. This image contains for the most part black text on a

white background. In average PNG performs the best, although this is solely due to the very large compression ratio it achieves on “target”. However, JPEG-LS provides the best compression ratio for most images.

This shows that as far as lossless compression is concerned, JPEG 2000 seems to perform reasonably well in terms of its ability to efficiently deal with various types of images. However, in specific types of images such as “cmpnd1” and “target” JPEG 2000 is outperformed by far in JPEG-LS. This result is even more striking noting that JPEG-LS is a significantly less complex algorithm.

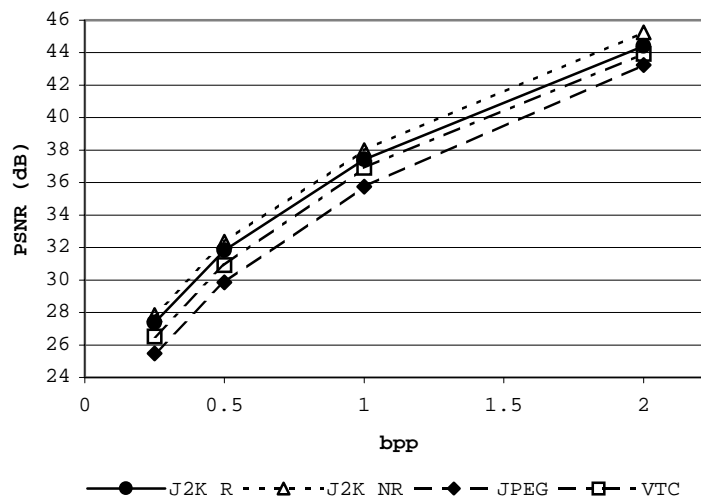
Table 2 shows the execution times, relative to JPEG-LS, for decompression. It shows that JPEG-LS, in addition to providing the best compression ratios, is close to the fastest algorithm, L-JPEG, and therefore presumably of low complexity. JPEG 2000 is considerably more complex while PNG is close to JPEG-LS. It should be noted that while JPEG-LS and JPEG 2000 are symmetrical (i.e. encoding and decoding times are similar), it is not the case for PNG, which is strongly asymmetrical (i.e. encoding time is much longer than decoding time). L-JPEG, as used when generating these results, is asymmetrical, although it can be made symmetrical if the default Huffman tables and predictor are used, at the expense of degraded compression efficiency.

**Table 2.** Lossless decoding times, relative to JPEG-LS, and JPEG-LS absolute times in secs.

	JPEG 2000 <sub>R</sub>	JPEG-LS	L-JPEG	PNG	JPEG-LS abs (secs.)
bike	3.7	1.0	0.7	0.9	2.17
cafe	4.0	1.0	0.7	1.0	2.25
cmpnd1	6.7	1.0	1.6	2.2	0.06
chart	4.0	1.0	0.9	1.2	1.2
aerial2	4.1	1.0	0.7	0.8	1.76
target	3.9	1.0	0.9	1.3	0.07
us	3.9	1.0	0.7	1.1	0.07
<b>average</b>	<b>4.3</b>	<b>1.0</b>	<b>0.9</b>	<b>1.2</b>	-

#### 4.2. Fixed lossy compression

Figure 1 shows the average rate distortion behavior when fixed (i.e. non-progressive) compression is used, for the evaluated schemes that are capable of lossy compression. For each bitrate, a single bitstream is generated and decoded in its entirety. For JPEG 2000, results for reversible and non-reversible DWT filters are shown, referred to as J2K<sub>R</sub> and J2K<sub>NR</sub> respectively. For MPEG-4 VTC the results have been generated using single quantization (SQ). In the case of JPEG the baseline mode has been used with flat quantization tables and optimized Huffman tables. The results shown are also typical, for all the tested images.



**Figure 1.** PSNR corresponding to average RMSE, of all test images, for each algorithm when performing lossy coding at 0.25, 0.5, 1 and 2 bpp.

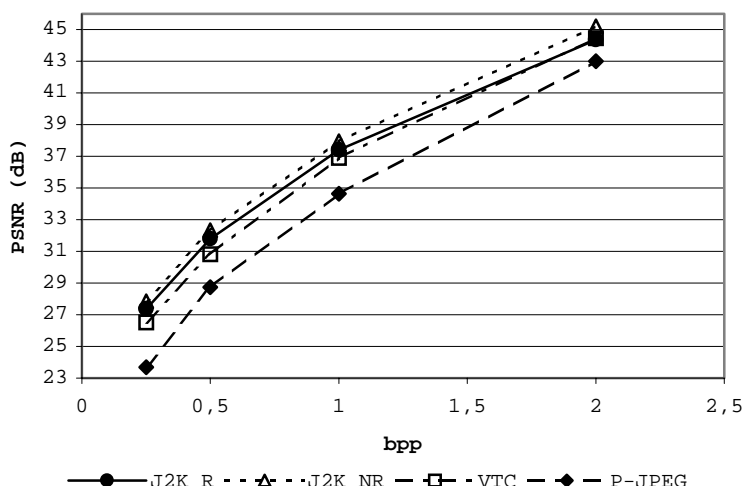
As it can be seen, JPEG 2000 outperforms all other algorithms. The non-reversible filter provides higher compression efficiency than the reversible one, but is not capable of lossless coding. JPEG provides, as expected for older technology, inferior results, showing a considerable quality difference at any given bitrate. MPEG-4 VTC provides results in between JPEG and JPEG 2000.

### 4.3. Progressive lossy compression

Figure 2 depicts the average rate-distortion behavior obtained by applying progressive compression schemes studied in this paper on the set of tested images. For JPEG 2000, results for reversible and non-reversible DWT filters are shown, referred to as J2K<sub>R</sub> and J2K<sub>NR</sub> respectively, with a SNR scalable bitstream. For MPEG-4 VTC the results have been generated using multiple quantization (MQ) and targeting the tested bitrates. In the case of JPEG the progressive mode has been used where the coefficients are encoded by successive refinement, and is referred to as P-JPEG. The results shown are also typical, for all the tested images.

As it can be clearly seen, progressive lossy JPEG 2000 outperforms all other schemes. The progressive lossless JPEG 2000 does not perform as well, mainly due to the use of reversible wavelet filters, however a lossless version of the image remains available after compression (not shown in the figure), which can be of significant value to many applications (archiving, medical, etc.). MPEG-4 VTC provides results comparable to those of JPEG 2000 with the reversible filter at high bitrates, although at lower ones a difference appears. As for progressive JPEG, it is outperformed by far by the other algorithms.

It is also worth noting that when the above results are compared to the non-progressive ones shown above it is clearly seen that JPEG 2000's and MPEG-4 VTC's compression performance is not adversely affected when SNR scalable bitstreams are generated. On the contrary, this cannot be said about JPEG.



**Figure 2.** PSNR corresponding to average RMSE, of all test images, for each algorithm when performing lossy decoding at 0.25, 0.5, 1 and 2 bpp of the same progressive bitstream encoded at 2 bpp.

Table 3 shows the execution times, relative to P-JPEG, for decompression. It shows that JPEG is the fastest algorithm and therefore presumably the least complex. JPEG 2000 is significantly slower, but provides extra compression performance. As it can be seen, the use of reversible filters is considerably faster than non-reversible ones, due to the simpler (5,3) filter and the use of integer arithmetic instead of the slower floating-point one. A surprise is MPEG-4 VTC, which is extremely slow compared to the other algorithms, without providing any better compression than JPEG 2000. This could be due to badly written software and these numbers should be considered as a rough indicative measure only. In contrast to the lossless case, the algorithms here are roughly symmetrical.

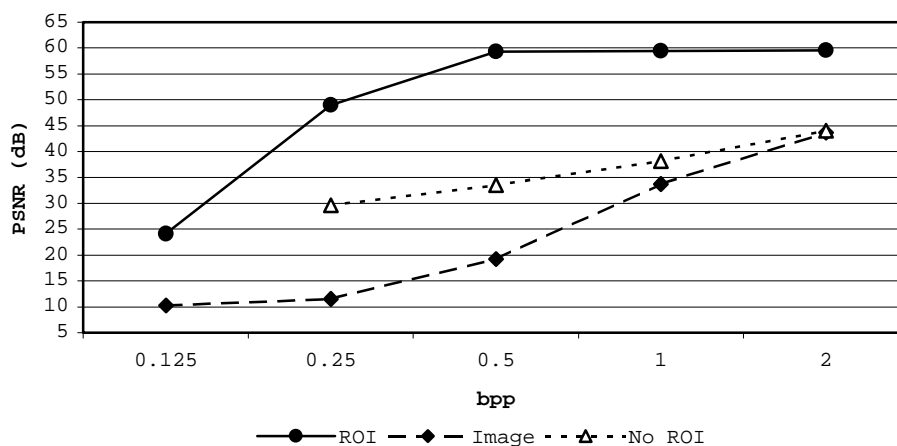
**Table 3.** Lossy decoding times of a progressive bitstream at 2 bpp, relative to P-JPEG, and P-JPEG absolute times in secs.

	JPEG 2000 <sub>R</sub>	JPEG 2000 <sub>NR</sub>	P-JPEG	MPEG-4 VTC	P-JPEG abs (secs.)
bike	2.8	3.6	1.0	35.1	1.79
cafe	2.8	3.6	1.0	36.1	1.76
cmpnd1	3.2	3.9	1.0	36.4	0.12
chart	2.8	3.6	1.0	34.6	1.32
aerial2	2.6	3.6	1.0	35.6	1.46
target	3.2	4.1	1.0	36.3	0.08
us	3.0	4.0	1.0	35.6	0.07
<b>average</b>	<b>2.9</b>	<b>3.8</b>	<b>1.0</b>	<b>35.7</b>	-

#### 4.4. ROI coding

Figure 3 shows the PSNR calculated using the average RMSE values for the same set of test images as were used previously. The PSNR values were calculated for both the ROI and the entire image and a comparison was made with the results for progressive decoding of non-reversible bit-streams obtained in section 4.2. The Region of Interest in each case was a rectangular area with its upper left corner at (0.5, 0.5) times the image dimensions and the width and height of the ROI were 0.25 and 0.25 of the image dimension, respectively. A single non-reversible SNR progressive bit-stream was generated for each image and it was then decoded at 0.125, 0.25, 0.5, 1.0 and 2.0 bits per pixel.

The graph clearly shows that the quality of the ROI is significantly better than the quality of the image encoded without an ROI, particularly at low bit rates. The graph also shows that the quality of the ROI and of the rest of the image converges and they will both eventually reach the same quality when the full bit-stream has been decoded. Note that for the images tested this occurs at bitrates higher than 2 bpp.



**Figure 3:** PSNR corresponding to average RMSE, of all test images, for ROI and the entire image (Image) and for an image without ROI (No ROI) when performing lossy decoding at 0.125, 0.25, 0.5, 1 and 2 bpp of the same progressive bitstream.

#### 4.5. Error resilience

In order to evaluate the error resilience features offered by the different standards, we have simulated a symmetric binary transmission channel with random errors and evaluated the average reconstructed image quality after decompression. Table 4 shows the results for JPEG 2000, with the reversible and non-reversible filters, and JPEG baseline. JPEG-LS and MPEG-4 VTC could not be evaluated since the software did not offer proper error resilience support. Concerning PNG, the comparison is not applicable, since this format only supports error detection, not concealment. Due to lack of space only the results of the cafe image are shown, however, the behavior is very similar for the other images. In the case of JPEG the results have been obtained by using the maximum amount of restart markers, which amounts to an overhead of less than 1%. In the case of JPEG 2000 the sensitive packet head information has been moved to the bitstream header (using a PPM

marker) and the entropy coded data has been protected by the regular termination of the arithmetic coder combined with the error resilient termination and segment symbols. The overhead of these protections amount also to less than 1%. In both cases the bitstream header is transmitted without errors.

**Table 4.** PSNR, in dB, corresponding to average MSE, of 200 runs, of the decoded “cafe” image when transmitted over a noisy channel with various bit error rates (BER) and compression bitrates, for JPEG baseline and JPEG 2000, with reversible and non-reversible filters (JPEG 2000<sub>R</sub> and JPEG 2000<sub>NR</sub>, respectively).

BER	JPEG 2000 <sub>R</sub>				JPEG 2000 <sub>NR</sub>				JPEG			
	0.25	0.5	1.0	2.0	0.25	0.5	1.0	2.0	0.25	0.5	1.0	2.0
0	22.64	26.21	31.39	38.27	23.06	26.71	31.91	38.93	21.94	25.39	30.34	37.23
1e-6	22.45	26.01	30.25	36.06	22.99	26.20	29.70	34.85	21.77	25.11	29.18	28.29
1e-5	20.37	23.35	25.91	25.80	21.11	23.06	25.8	24.68	20.42	22.61	22.33	19.67
1e-4	16.02	16.20	16.52	17.16	16.14	16.57	16.29	16.71	16.16	15.38	14.49	12.02

As it can be seen, the reconstructed image quality under transmission errors is higher for JPEG 2000 than JPEG, across all encoding bitrates and error rates. Although both suffer from severe degradation at moderately high error rates (i.e. 1e-4 and 1e-5), at lower ones (i.e. 1e-6) JPEG 2000 proves to be fairly robust. Also the visual quality of JPEG 2000 at these lower error rates is much higher than that of JPEG. In fact, the artifacts created by transmission errors under JPEG 2000 are of the same nature as those created by quantization. In the case of JPEG, when a transmission error occurs it is often entire 8x8 blocks that will be missing and/or misplaced and the bottom of the image will often be missing as well.

It should also be noted that at higher error rates (i.e. 1e-4), the reconstructed image quality in JPEG 2000 increases very little with increasing bitrate. This is due to the fact that in JPEG 2000 each sub-band block is coded by bitplanes. When the error rate is high enough almost all code-blocks are affected in the most significant bitplanes, which are transmitted first. When a particular bitplane is affected in a block, lower bitplanes cannot be decoded and are therefore useless. In the case of JPEG the problem is even worse: the higher the encoding bitrate the lower the decoded quality. This can be explained by the fact that in JPEG the error is at the block level at the most and therefore the density of error protection decreases with an increase in bitrate.

#### 4.6. Functionality

Table 5 summarizes the results of the comparison of different algorithms from a functionality point of view. The table clearly shows that from this perspective, JPEG 2000 is the standard offering the richest set of features in an efficient manner and within an integrated algorithmic approach. In this table we refer to *genericity*, which is the ability to efficiently compress different types of imagery across a wide range of bitrates.

**Table 5.** Functionality matrix. A “+” indicates that it is supported, the more “+” the more efficiently or better it is supported. A “-” indicates that it is not supported.

	JPEG 2000	JPEG-LS	JPEG	MPEG-4 VTC	PNG
lossless compression performance	+++	++++	+	-	+++
lossy compression performance	+++++	+	+++	++++	-
progressive bitstreams	+++++	-	++	+++	+
Region of Interest (ROI) coding	+++	-	-	+	-
arbitrary shaped objects	-	-	-	++	-
random access	++	-	-	-	-
low complexity	++	+++++	+++++	+	+++
error resilience	+++	++	++	+++	+
non-iterative rate control	+++	-	-	+	-
genericity	+++	+++	++	++	+++

MPEG-4 VTC, as JPEG 2000, is able to produce progressive bitstreams without any noticeable overhead. However, the latter provides more progressive options and produces bitstreams that are parseable and that can be rather easily reorganized by a transcoder on the fly. Along the same lines, JPEG 2000 also provides random access (i.e. involving a minimal decoding) to the block level in each sub-band, thus making possible to decode a region of the image without having to decode it as a whole. These two features could be very advantageous in applications such as digital libraries.



Another result of the fact that JPEG2000 generates progressive bitstream is the Region of Interest functionality. As shown in the previous section it is possible to generate a bitstream in which specific areas of the image are refined earlier than the rest. The bitstream can be organized so that either the image is stored with the same quality for the ROI and the rest of the image, or if the bitstream is truncated, the ROI will have better quality than the background.

Concerning error resilience JPEG 2000 offers higher protection than JPEG, as shown in the previous section. MPEG-4 VTC also offers error resilience features and although it could not be evaluated, the support is expected to be in between JPEG and JPEG 2000. JPEG-LS does not offer any particular support for error resilience, besides restart markers, and has not been designed with it in mind. PNG offers error detection, but no concealment possibilities.

Overall, one can say that JPEG 2000 offers the richest set of features and provides superior rate-distortion performance. However, this comes at the price of additional complexity when compared to JPEG and JPEG-LS, which might be currently perceived as a disadvantage for some applications, as was the case for JPEG when it was first introduced.

## 5. CONCLUSIONS

This work aims at providing a comparison of the efficiency of various features that can be expected from a number of recent as well as most popular still image coding algorithms. To do so, many aspects have been considered including genericity of the algorithm to code different types of data in lossless and lossy way, and features such as error resiliency, complexity, scalability, region of interest, embedded bitstream and so on.

The results presented in previous sections show that from a functionality point of view JPEG 2000 is a true improvement, providing lossy and lossless compression, progressive and parseable bitstreams, error resilience, region of interest, random access and other features in one integrated algorithm. However, while new standards provide higher compression efficiency there is no truly substantial improvement. This is especially true for lossy coding, even though the new standards, except for JPEG-LS, are significantly more complex than JPEG.

In any case, the choice of a standard for a particular application or product will depend on its requirements. In the cases where lossy compression is of interest and low complexity is of high priority, JPEG still provides a good solution. JPEG-LS stands out as the best option when only lossless compression is of interest, providing the best compression efficiency at a low complexity. PNG is also of interest in such cases, although the complexity of the encoder is much higher than that of JPEG-LS. As for MPEG-4 VTC, it appears to be of limited interest, except when the ability to code arbitrarily shaped objects is required. JPEG 2000 provides the most flexible solution, combining good compression performance with a rich set of features.

## 6. REFERENCES

1. ISO/IEC JTC1/SC29/WG1, *Call for contributions for JPEG 2000 (JTC 1.29.14, 15444): Image Coding System*, WG1 N 505, March 1997, <http://www.jpeg.org/public/wg1n505.pdf>.
2. ISO/IEC JTC 1/SC 29/WG 1, *ISO/IEC FCD 15444-1: Information technology — JPEG 2000 image coding system: Core coding system [WG 1 N 1646]*, March 2000, <http://www.jpeg.org/FCD15444-1.htm>.
3. William B. Pennebaker and Joan L. Mitchell, *JPEG: Still Image Data Compression Standard*, Van Nostrand Reinhold, New York, 1992.
4. ISO/IEC, *ISO/IEC 14496-2:1999: Information technology — Coding of audio-visual objects — Part 2: Visual*, December 1999.
5. ISO/IEC, *ISO/IEC 14495-1:1999: Information technology — Lossless and near-lossless compression of continuous-tone still images: Baseline*, Dec. 1999.
6. W3C, *PNG (Portable Network Graphics) Specification*, Oct. 1996, <http://www.w3.org/TR/REC-png>.
7. ISO/IEC, *ISO/IEC 11544:1993 Information technology — Coded representation of picture and audio information — Progressive bi-level image compression*, March 1993.
8. ISO/IEC JTC 1/SC 29/WG 1, *ISO/IEC FCD 14492: Information technology — Coded representation of picture and audio information — Lossy/Lossless coding of bi-level images [WG 1 N 1359]*, July 1999, <http://www.jpeg.org/public/jbigpt2.htm>.
9. D. Le Gall and A. Tabatabai, "Sub-band coding of digital images using symmetric short kernel filters and arithmetic coding techniques", *Proc. Of the IEEE International Conference on Acoustics, Speech and Signal Processing*, New York, NY, USA, pp. 761-765, 1988.

In Proc. of the SPIE's 45th annual meeting, Applications of Digital Image Processing XXIII, volume 4115, pages 446-454, San Diego, California, Jul. 30-Aug. 4, 2000, <http://ltswww.epfl.ch/~dsanta>.

10. M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using the wavelet transform", *IEEE Trans. On Image Processing*, vol. 1, pp. 205-220, April 1992.
11. C. Christopoulos, J. Askelöf, M. Larsson, "Efficient methods for encoding Regions of Interest in the upcoming JPEG 2000 still image coding standard", *IEEE Signal Processing Letters*, September 2000.