In Proc. of the X European Signal Processing Conference, vol. 2, pp. 673-676, Tampere, Finland, September 5-8, 2000, `http://ltswww.epfl.ch/~dsanta`.

A STUDY OF JPEG 2000 STILL IMAGE CODING VERSUS OTHER STANDARDS

Diego Santa-Cruz and Touradj Ebrahimi
Signal Processing Laboratory, Swiss Federal Institute of Technology, Lausanne
LTS / DE / EPFL, CH-1015 Lausanne, SWITZERLAND
Tel: +41 21 6934620; Fax: +41 21 6937600
e-mail: Diego.SantaCruz@epfl.ch, Touradj.Ebrahimi@epfl.ch

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 puts into perspective the performance of these by evaluating JPEG 2000 versus JPEG-LS and MPEG-4 VTC, as well as the older but widely used JPEG. The study concentrates on compression efficiency, although complexity and set of supported functionalities are also evaluated. Lossless compression efficiency as well as the lossy rate-distortion behavior is discussed. The principles behind each algorithm are briefly described and an outlook on the future of image coding is given. The results show that the choice of the "best" standard depends strongly on the application at hand.

1 INTRODUCTION

It has been three years since the call for proposals [4] 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) [3] 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: How good is JPEG 2000 when compared with other popular still image coding techniques? This paper aims at providing an answer to this trivial but somewhat complex question. The paper compares JPEG 2000 to other coding standards on the basis of compression efficiency as well as functionality set, and provides an outlook on the future of image coding. Section 2 provides a brief overview of the techniques compared. Section 3 explains the comparison methodology employed in the results shown in section 4. Section 5 provides a look into the future of image coding and conclusions are drawn in section 6.

2 OVERVIEW OF EVALUATED STANDARDS

This paper compares JPEG 2000 to the following three standards: JPEG [5], MPEG-4 Visual Texture Coding (VTC) [2] and JPEG-LS [1]. The last two have been chosen because they are new standards that start appearing in various applications and represent the state-of-the-art in image coding, while the first is the most widely used. In the following a brief explanation of the principles behind the algorithms used in each of these standards is given.

2.1 JPEG

This is the very well known ISO/ITU-T standard created in the late 1980s. There are several modes defined in JPEG, however, here we refer to only two: baseline and lossless. 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 other modes defined in JPEG provide variants of the previous two basic modes, such as progressive bitstreams and arithmetic entropy coding. One such other mode, which is increasingly popular, is progressive JPEG, in which the quantized samples are sent progressively by a mixture of spectral selection and successive approximation.

2.2 MPEG-4 VTC

MPEG-4 Visual Texture Coding (VTC) is the algorithm used in MPEG-4 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, zerotree coding and arithmetic coding. The DWT is dyadic. 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. VTC supports coding of arbitrarily shaped objects, by the means of a shape adaptive DWT, but does not support lossless coding.

2.3 JPEG-LS

JPEG-LS is the latest ISO/ITU-T standard for lossless coding of still images. It also provides for "near-lossless" compression. It 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. 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 JPEG 2000

JPEG 2000, 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 a reversible filter, which provides for lossless coding, or a non-reversible one, 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 blocks, 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 error-resilience, arbitrarily shaped region of interest, 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, concern 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. In this paper we concentrate in compression efficiency, since it is still one of the top priorities in many imaging products, but we also devote attention to complexity and functionalities.

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 $-20\log_{10}(\mathrm{RMSE}/(2^b-1))$, 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.

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 workstation. This only gives an appreciation of the involved complexity.

3.3 Functionalities

Most applications necessitate other features in a coding algorithm than simple compression efficiency. This is often referred to as functionalities. In the next section we provide a functionality matrix which indicates the set of supported features in each standard and an appreciation of how well they are fulfilled.

4 RESULTS

The results have been generated on a PC with a 550 MHz Pentium$^{\mathrm{TM}}$ III processor, 512 kB of cache and 512 MB of RAM under Linux 2.2.12. The softwares 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 Lossless JPEG codec of Cornell University (ftp://ftp.cs.cornell.edu/pub/multimed), version 1.0, and the SPMG JPEG-LS implementation of the University of British Columbia (http://spmg.ece.ubc.ca), version 2.2.

Table 1: Lossless compression ratios.

	$J2K_R$	JPEG-LS	L-JPEG	BZIP2
bike	1.77	1.84	1.61	1.72
cafe	1.49	1.57	1.36	1.40
cmpnd1	3.77	6.44	3.23	5.65
chart	2.60	2.82	2.00	2.39
aerial2	1.47	1.51	1.43	1.63
target	3.76	3.66	2.59	7.34
us	2.63	3.04	2.41	3.22
average	**2.50**	**2.98**	**2.09**	**3.34**

The images used are 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.

4.1 Lossless compression

Here the JPEG 2000, JPEG-LS and L-JPEG algorithms have been tested. MPEG-4 VTC has been omitted because it does not support lossless coding. In the case of JPEG 2000 the reversible DWT filter, which is referred to as $J2K_R$, 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. In addition, the results of the general purpose "bzip2" compressor commonly found on recent UNIX machines are also reported (version 0.9.5c with the maximum compression level has been used).

Table 1 shows the lossless compression ratios obtained for each image, as well as the average for each algorithm. It can be seen that JPEG-LS provides the best compression ratios for almost all images, while JPEG 2000 achieves competitive results. One exception is the "cmpnd1" image, in which JPEG-LS achieves much larger compression. This image contains, for the most part, black text on a white background, which statistics are best captured by JPEG-LS. As for L-JPEG, the compression performance is not very good and shows a lower adaptability to different image types. One surprise is BZIP2, which achieves impressive results given the fact that it does not take into account that the data represent images. This is specially true for synthetic images, specially "target" which contains mostly patches of constant grey level as well as gradients. In average BZIP2 performs the best. However, this is solely due to the very large compression ratio it achieves on "target". Generally speaking, JPEG-LS provides the best compression ratios.

Table 2 shows the execution times, relative to JPEG-LS, for compression. It shows that JPEG-LS, in addition to providing the best compression ratios, is the fastest algorithm, and therefore presumably the least complex. JPEG 2000 is considerably more complex. L-JPEG is in between, but does not provide any compression efficiency advantage.

Table 2: Lossless encoding times, relative to JPEG-LS, and JPEG-LS absolute times in secs.

	$J2K_R$	L-JPEG	BZIP2	JPEG-LS abs.
bike	4.6	1.9	4.7	2.01 secs.
cafe	4.9	1.9	5.4	2.08 secs.
cmpnd1	7.1	3.6	3.4	0.07 secs.
chart	5.1	2.5	4.9	1.09 secs.
aerial2	4.8	1.9	4.6	1.64 secs.
target	5.5	2.8	5.0	0.06 secs.
us	5.3	1.0	3.8	0.06 secs.
average	**5.3**	**2.2**	**4.6**	-

4.2 Lossy compression

Here the JPEG 2000, baseline JPEG, and MPEG-4 VTC algorithms have been tested at bitrates of 0.25, 0.5, 1.0 and 2.0 bits per pixel (bpp). For each, one bitstream is created, fully decoded and the distortion of the reconstructed image evaluated by means of the RMSE. Although JPEG-LS supports lossy compression it has been designed for lossless and "near-lossless" only and is not suited for running at the tested bitrates. This is why it is not included in the lossy compression comparisons.

For JPEG 2000 a resolution progressive bitstream is created using both the reversible and non-reversible DWT filters, referred to as $J2K_R$ and $J2K_{NR}$, respectively. For JPEG flat quantization tables and optimized Huffman tables are used to improve the RMSE. For MPEG-4 VTC single quantization is used, generating a non-scalable bitstream.

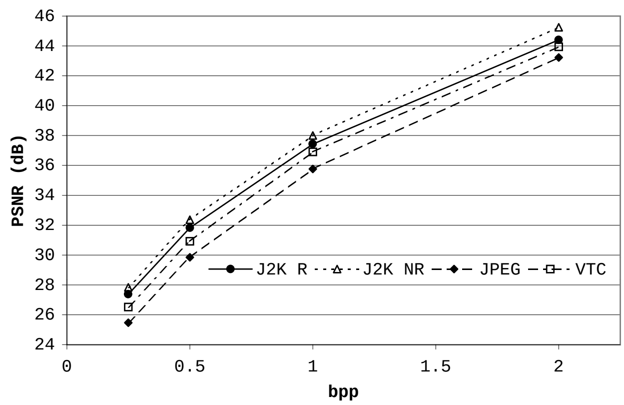

Figure 1: PSNR corresponding to average RMSE, of all test images, for each algorithm when performing lossy encoding at 0.25, 0.5, 1 and 2 bpp.

Figure 1 depicts the average rate-distortion behavior of each algorithm. As it can be seen, JPEG 2000 outperforms all other algorithms; the non-reversible filter provides higher compression efficiency when compared to the reversible, but with the latter it is possible to perform a lossless decoding (not shown in the figure). JPEG provides, as expected for older technology, inferior results, showing a considerable quality difference at any given bitrate. MPEG-4 VTC provides results that are in between JPEG and JPEG 2000.

Table 3 shows the execution times, relative to JPEG, for

Table 3: Lossy encoding times at 2 bpp, relative to JPEG, and JPEG absolute times in secs.

	$J2K_R$	$J2K_{NR}$	VTC	JPEG abs.
bike	5.3	5.9	17.2	1.67 secs.
cafe	5.9	6.5	16.9	1.7 secs.
cmpnd1	3.6	4.4	13.5	0.14 secs.
chart	4.5	5.0	15.5	1.22 secs.
aerial2	5.4	6.1	16.9	1.41 secs.
target	3.7	4.3	13.6	0.09 secs.
us	4.0	4.8	13.6	0.08 secs.
average	**4.6**	**5.3**	**15.3**	-

compression. It shows that JPEG is the fastest algorithm and therefore presumably the least complex. JPEG 2000 is considerably slower, but provides extra compression performance. 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.

4.3 Functionalities

Table 4 summarizes the comparison from a functionality point of view. This clearly shows that JPEG 2000 is the standard offering the richest set of features in an efficient manner and integrated algorithmic approach. However, MPEG-4 VTC is the only standard supporting the coding of arbitrary shaped objects.

5 FUTURE OF IMAGE CODING

The results in the previous section show that there have not been any truly significant advancements in image compression efficiency in the past decade, despite the intensive research activity of the field. Commonly used image sources (scanners, digital cameras, etc.) provide a pixel-based representation of image data, which is kept by the compression algorithms. However, there is an increasing number of computer generated images which are not originally pixel-based, but are currently converted to such a representation prior to compression. New compression algorithms could work on the specific nature of these images and keep a representation based on other mathematical constructs. While compression of pixel-based images appears to reach a limit, this new approach could open the door for increased compression efficiency for computer generated images while having the advantage of being resolution independent.

6 CONCLUSIONS

The results presented in previous sections show that new standards do not provide any truly substantial improvement in compression efficiency and are significantly more complex than JPEG, with the exception of JPEG-LS for lossless compression. However, from a functionality point of view JPEG 2000 is a true improvement, providing lossy and lossless compression, progressive and parseable bitstreams, error

Table 4: 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.

	J2K	JPEG-LS	JPEG	VTC
lossless comp.	+++	++++	+ [a]	-
lossy comp.	+++++	+	+++	++++
progressive	++++	-	+ [b]	++
ROI[c] coding	+++	-	-	+ [d]
arbitrary shaped objects	-	-	-	++
random access	++	-	-	-
low complexity	++	+++++	+++++	+
error resilience	+++	+	+	+++
non-iterative rate control	+++	-	-	+
genericity[e]	+++	+++	++	++

[a] Only using the lossless mode of JPEG.
[b] Only in the progressive mode of JPEG.
[c] Region Of Interest.
[d] Tile-based only.
[e] Ability to efficiently compress different types of imagery across a wide range of bitrates.

resilience, random access, region of interest and other features in one integrated algorithm.

In any case the choice of a standard for a particular application or product will depend on its requirements. JPEG-LS stands out as the best option when only lossless compression is of interest, providing the best compression efficiency at a low complexity. In the cases where lossy compression is of interest and low complexity is of high priority JPEG still provides a good solution. On the other hand JPEG 2000 provides the most flexible solution, if the added complexity is acceptable. As for MPEG-4 VTC, it appears to be of limited interest, except when the ability to code arbitrary shaped objects is required.

References

[1] ISO/IEC, ISO/IEC 14495-1:1999: Information technology — Lossless and near-lossless compression of continuous-tone still images: Baseline, December 1999.

[2] ISO/IEC, ISO/IEC 14496-2:1999: Information technology — Coding of audio-visual objects — Part 2: Visual, December 1999.

[3] 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.

[4] ISO/IEC JTC1/SC29/WG1, Call for contributions for JPEG 2000 (JTC 1.29.14, 15444): Image Coding System, WG1 N 505, March 2000.

[5] W. B. Pennebaker and J. L. Mitchell, JPEG: Still Image Data Compression Standard. New York: Van Nostrand Reinhold, 1992.