

JPEG 2000 standardization - a pragmatic viewpoint

Richard Clark

JPEG webmaster, UK head of delegation
Editor, JPEG 2000 Reference Software (etc.)

Elysium Ltd



+44 1892 667411 - info@elysium.ltd.uk

Today's presentation

- ❖ Cover a bit of background - rationale and progress on JPEG 2000
- ❖ Talk a little about what aspects of the profile are important, and what they mean
- ❖ Take an example and see how changes can affect how an image might be delivered
- ❖ Colour metrics and their importance
- ❖ Viewer programs
- ❖ This is a short presentation - there is LOTS more



indicates ask me afterwards!

What do we need from an imaging standard?

❖ From an archival and preservation perspective

- Stability
- Reliability
 - Standard should not lead to unconforming variants
- Ease of use
- Flexibility
 - Ability to fit into a workflow, and to be able to generate wide variety of output formats
- Low cost of implementation
- Widespread support
- Fully documented
- Quality

What do we need from an imaging standard?

❖ Original JPEG

- **Stability** - In use for 20 years - original files still readable, but some information missing (e.g. colourspace, or artefacts from compression)
- **Reliability** - Very unusual to find JPEG images that are not fully portable
- **Ease of use** - Tick
- **Flexibility** - Not very. Cannot combine lossy/lossless. No resolution/quality trade-off. File formats (e.g. metadata) have a lot of options
- **Low cost of implementation** - Tick. High quality, Open Source available, few patent issues (and now hopefully old enough to avoid future ones!) 
- **Widespread support** - Tick. In browsers, viewers etc - de facto.
- **Fully documented**- Tick.
- **Quality** - Can do better (however can also do better with existing JPEG too) 

What do we need from an imaging standard?

❖ JPEG 2000

- **Stability** - 10 years old
- **Reliability** - In general not a problem - some implementations can be flaky (e.g. with huge images) 
- **Ease of use** - So many options , can be confusing, especially as sold in toolkit form
- **Flexibility** - Designed for full flexibility, as an architecture. Can align many components, resolutions, qualities in a single file
- **Low cost of implementation** - OK, relative to costs of scanning, storing and indexing is tiny - hopefully paid for with efficiency gain
- **Widespread support** - Getting better. Still missing in most browsers (and not good in the ones its in!). Variety of toolkits now available 
- **Fully documented** - Yes, and books starting to improve that situation 

JPEG v JPEG 2000 (v JPEG LS v JBIG v JPEG XR)

JPEG - Pretty, but old,
and fixed



JPEG 2000- a toolkit. But you have to build from it!
You can build what you want. It --should-- be
better, but it depends on the engineer!



JPEG LS - Does what it says very well. Fast,
but restricted



JBIG - Best compression for bi-tonal. Can
be used in combination with J2K



JPEG XR - If you want Microsoft's best..



Original J2K objectives



- ❖ "The goal of the contribution process is to gather algorithms, components of algorithms, and architectural frameworks; and to organize algorithm components into a single architecturally based standard. An architecturally based standard has the potential of allowing the JPEG 2000 standard to evolve and integrate new algorithm components without requiring (*sic*) a new standards definition"
- ❖ "An important component of committee participation is the understanding of the work of other contributors and the participation in the convergence toward a standard that takes advantage of the best of all the contributions. "
- ❖ "Intellectual property rights are a major concern in this activity. The ISO requires that any contributor offers a license of the relevant intellectual property to all without favor for reasonable rates. Contributors should evaluate what their license compensation needs are before contributing to this standard"

Original intent

- ❖ Document imaging
- ❖ Financial documents
- ❖ Facsimile
- ❖ Security Cameras
- ❖ Internet/WWW imagery
- ❖ Client-Server
- ❖ Laser print rendering
- ❖ Scanner/digital copiers
- ❖ Video component frames
- ❖ Prepress
- ❖ Photo and art digital libraries
- ❖ Electronic Photography
- ❖ Remote sensing
- ❖ Elevation
- ❖ Medical imagery
- ❖ Seismic

In practice, JPEG 2000 has been successful for security cameras, some specialist client server applications, in digital cinema (studio and cinemas), to some extent in medical imaging and in astronomy and archiving and preservation (not even on the list!).

It has been unsuccessful in the digital camera market, on the web as a delivery format, and has failed to make much impact in the professional scanning, print and pre-press arena. So - basically a toolkit , and can build excellent complex imaging applications

Current state of standard



Pt	Title	Published
1	JPEG 2000 Image Coding System: Core Coding System	00/12 Cons.04/09
2	JPEG 2000 Image Coding System: Extensions	01/11 Cons.04/05
3	Motion JPEG 2000	01/11 Cons.07/05
4	Conformance Testing	02/05
5	Reference Software	01/11
6	JPM: Compound Image File Format	03/04
7	--withdrawn--	
8	JPSEC: Secure JPEG 2000	06/07
9	JPIP: Interactivity tools, APIs and protocols	04/10
10	JP3D: Extensions for three dimensional data	07/12
11	JPWL: Wireless	07/04
12	ISO Media File Format	03/07
13	An entry level JPEG 2000 encoder	08/07
14	XML Structure Representation and Reference	11/11

Current state of standard ...

❖ Plus further 15 amendments (7 Part 1, 3 Part 2, plus 3/5/6/8/9)

❖ Majority to support digital cinema requirements (DCI)

- Joint venture with 7 major studios (now 6), created 2002
- open architecture, high level performance / quality
- keen to avoid MPEG volume based tariffs
- issued July 2005, updated 2007
- JPEG 2000 as video coder (uncompressed audio)
2048x1080 or 4096x2160 resolution, 12bit/component, XYZ
7.6:1 compression (2kx24), 5 level decomposition



❖ The point?

- The digital preservation and archiving community needs a voice and a direction!!
- The JPEG committee can (and I am sure will) help.....

All those parameters - what do they mean?

- ❖ As an example, the Kakadu JPEG 2000 s/w exposes more than most - you can change 106 parameters in the demo compression module alone!
- ❖ Don't worry - you don't need to. Most have sensible defaults, and they are there to allow profiles to be set up. Once you have that, you can forget them (hopefully!)
- ❖ At the other end of the spectrum, some applications try and make JPEG 2000 look like a JPEG compressor - Photoshop's plugin, or Apple's viewer in Safari. It very much misses the point...
- ❖ You need to worry about
 - What your original file size is, and how much you compress
 - What size the smallest image is likely to be needed
 - Can you vary the quality easily with resolution
 - Is it fast to transcode for delivery in other formats
 - Has it got all the metadata information required to image it correctly
 - Where you put it, and how to find it again

So let's see how it works....

Original from The National Archive

Using freely downloadable Kakadu toolkit (V 6.4.1) for testing/evaluation



338,641,348 bytes

9648 pixels wide by 11698 pixels high

Compressed with

- 7 quality layers
- 6 resolution levels (actually wavelet decompositions) - smallest is therefore $1/2^6$ (i.e. $1/64$ of the image resolution) - corresponds to lowest resolution level of 150 x 182
- Progression order either RLCP or LRCP - depends on need
- Compression in steps from 1.2bpp to 0.05 (20:1 to 480:1)
- Takes 25 secs on core 2 duo laptop, 2Ghz, and runs all available CPU, with a memory footprint up to c. 160MB



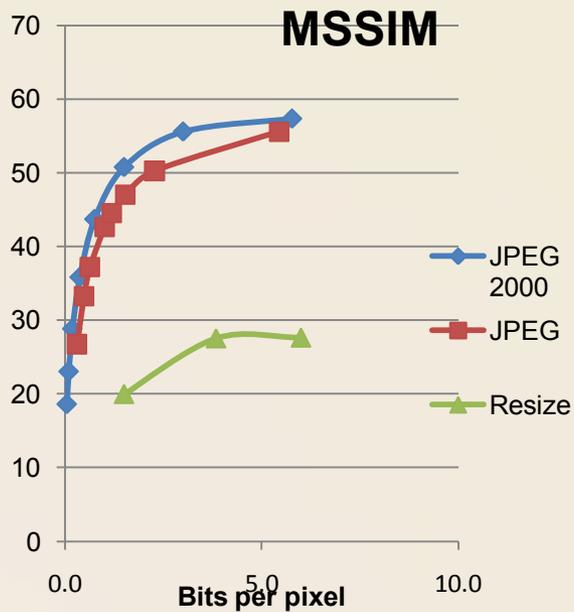
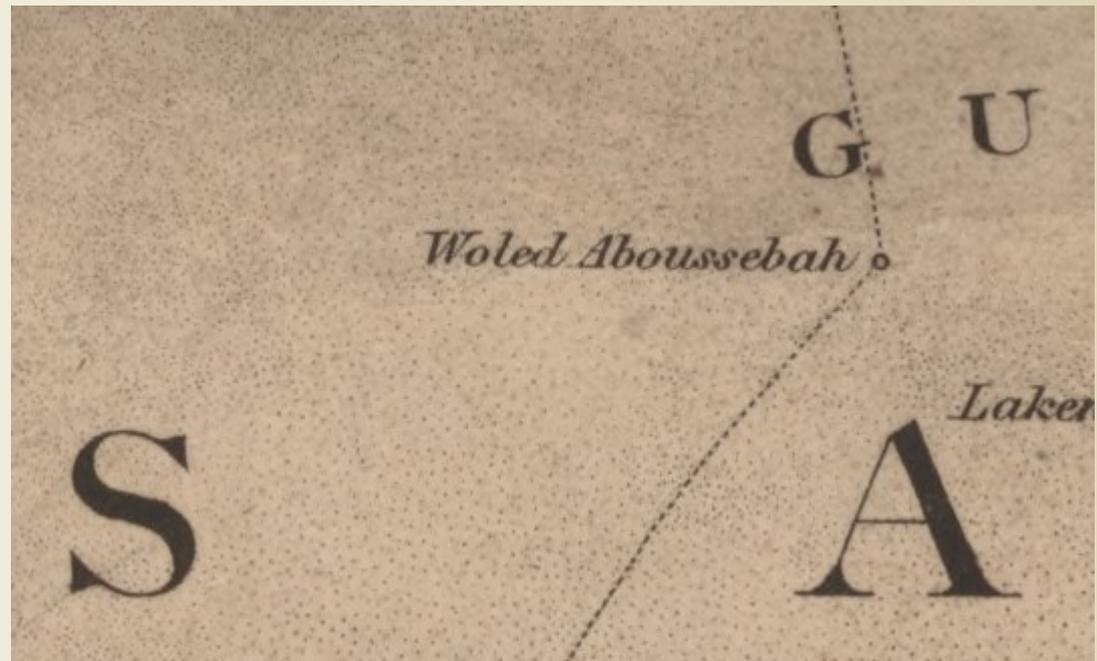
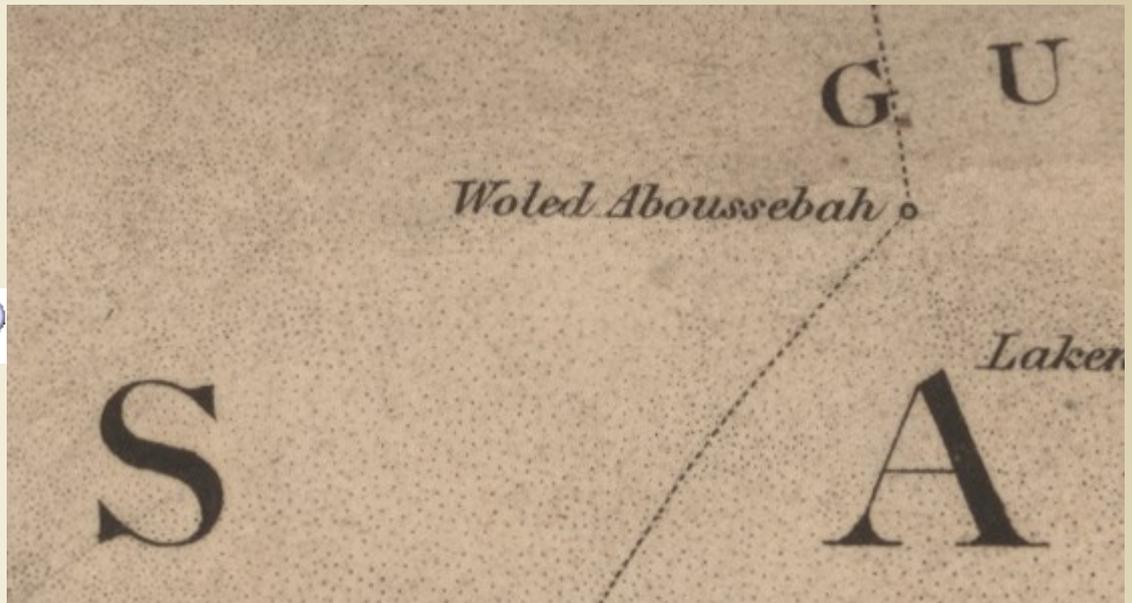
Results...

At 1:1 magnification, very little difference, even though this is at 20:1.

Most tests come up with value of c. 8:1 as visually lossless, and evidence this improves OCR for example.



There are effective measures of quality too - better than PSNR!



Now lets play...

Chop out c.100k bytes from front of file - very quick.

Run Kakadu transcode (near-instant) to fix file markers etc.

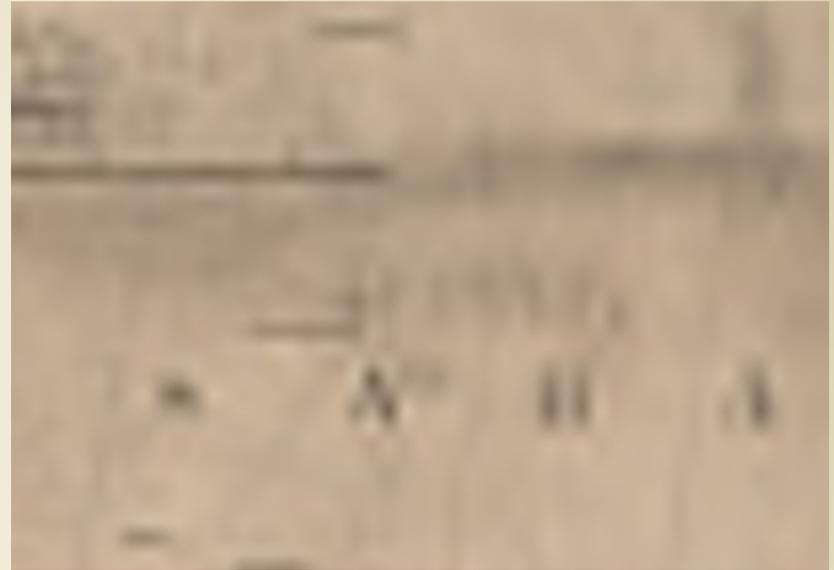


And here is the difference magnified between the first 100k (3,400:1 compression!) for RLCP (top) and for LRCP (bottom)

The version with LRCP is clearer (and looks visually better at a 16th of the original (basically at web resolution)

Transcoding from the higher quality image generated initially gives results comparable to the LRCP truncation, but doesn't care about the progression order, and takes about same time

?LRCP may be better if you are planning to get a fast variable size generation - for example for creating JPEG files on the fly, as a 'quick and dirty'



Just to confirm

At the Powerpoint resolution level, the map compressed at 3480:1 doesn't look a lot different from the original, although, as per the previous slide artefacts are very evident.

The power of JPEG 2000 is in its architecture, and the way that the coded data is built into packets that are easily rearranged and repurposed.

The progression order just make some actions easier/faster, especially for a server

- Layer
- Resolution
- Component (colour)
- Position (sometimes precinct)

Ignoring C and P, putting L first makes quality reduction trivial, and R first makes resolution reduction trivial (by truncating the code stream after the required number of bytes).

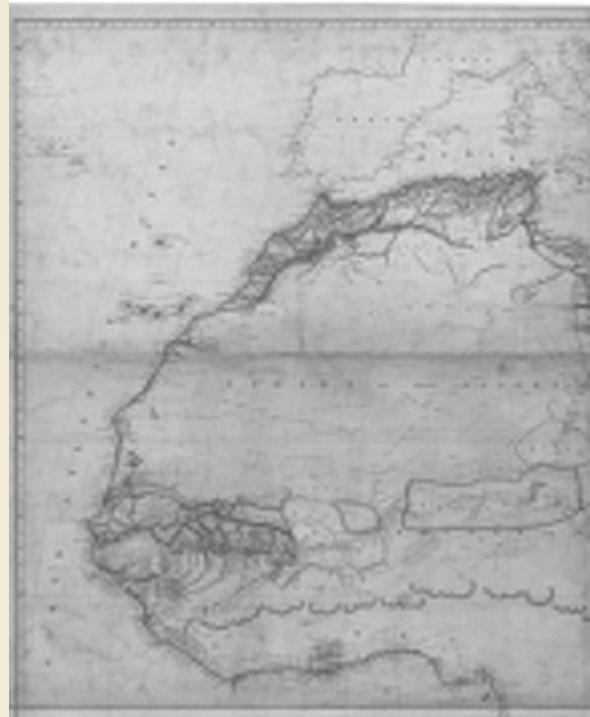


When do we really run out of data?

At the settings selected (7 layers, 6 levels) truncating to 40k bytes shows the component (colour) information starting to disappear. At 25k bytes, its all gone - but we --still-- have a thumbnail. Now RLCP is starting to look a little better in this test.

And a compression ratio of over 1300:1.....

Varying levels, layers and progression order will affect this end point. And of course, you don't just have to truncate the data stream (but it's --very-- fast)



First 25,000 bytes,
(fixed with
kdu_transcode)

First 40,000 bytes,
(fixed with
kdu_transcode).

Note: no component
information, bottom
right



ICC profile - support (at least under Windows)

See the test page for ICC V4 profiles at <http://www.color.org/version4html.xalter>



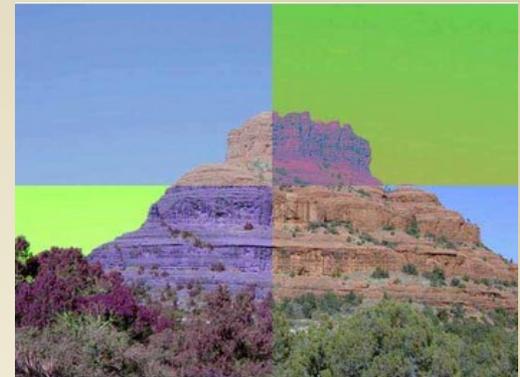
Although both Apple and Microsoft provide support for correct colour rendition through their respective engines, not everyone uses them. These are results for a variety of current browsers (Vista Home Premium as of Nov 2010).

However the trend is improving colour space support, and therefore from an archival perspective this is important.

However this is work in progress for JPEG 2000 - currently it too only indicates V2 profiles, unless the full profile is embedded and that requires a JPX file format (at the moment - watch this space...)



Firefox 3.6
ICC V2 compatible



Chrome V8 beta,
Internet Explorer 8
Opera 10.6.
No colour management



Safari 5.0
and correct (ICC V4 compat.)

Some JPEG 2000 viewer programs



- ❖ Kakadu (KDU_Show)
 - Good for testing and diagnosis - not great as a viewer, although supports JPIP and many other features. Very fast
 - ❖ Irfanview with a plugin
 - Slow
 - ❖ Quicktime viewer (also Preview on Snow Leopard == Kakadu V5.2.1)
 - Fairly basic, but proper colour space support? Crashes on large file (windows)
 - ❖ Aware viewer
 - Very slow
 - ❖ ER Viewer
 - From mapping specialist - fast - but a little awkward
 - ❖ Photoshop CS3
 - Not out of the box.
 - ❖ JHelioViewer (from NASA, apparently for astronomy only?)
 - Kakadu based, but much nicer. Java, very fast and Open Source. Try it!
- All of these are personal views, on Windows Vista. Your mileage may vary!*

What else should we think about?



- ❖ Integration into your workflow
 - How do you measure quality. How do you control and feedback in an automated conversion
- ❖ Metadata
 - Many different schemas , but how to keep them in synchronism, and linked to your DAM
- ❖ How to protect and deliver files
 - JPIP and alternatives. JPSEC can secure parts of files in different ways
- ❖ How to re-link files to their metadata (and the orphan works issue)
 - Image signatures (e.g. from the MPEG 7 toolkit, or via proprietary solutions) - should these be part of the metadata
- ❖ How to search and index the data
 - integration with legacy and other asset management tools
- ❖ File formats - all of the above
 - colour space, metadata, delivery, security
- ❖ Serving the files
 - Transcoding, formats, performance, storage
- ❖ Archiving files
 - Media life (DVD can be < 5 years), unique identifiers, hierarchy

Elysium and its goals.....



I guess I need to advertise... Standards work is UNPAID!

❖ Consultancy

- Advice and diagnosis of problems - prototyping, trials and tests

❖ Bespoke software design and implementation

- Skills vary from Open Source to Windows - languages include Java, PHP, C#, ASP.NET, Ruby/Rails, C++, Javascript and others, with design support too

❖ UK and European project work and partnership

- As partner in 6 European projects, 3 UK ones - all highly rated. We are always looking to increase our knowledge and develop new skills and products to commercialise

❖ Input to the standards work, especially JPEG

- Ow - the unpaid bit. 3 times a year, a week each time, around the world. Any suggestions as to how to keep this up and provide a UK input to get the J2K standards you need delivered are welcome

❖ Systems integration and project management

- These are major projects. Its important to get it right, on time, and on budget. We can help

Contact us at Elysium Ltd - see www.elysium.ltd.uk and demo.xalter.com for details

And input for the JPEG and ICC web sites is always appreciated - those are ours....



+44 1892 667411 - info@elysium.ltd.uk

And finally, to continue the theme....

Just to touch on patents, especially software patents.....

The Scene: Four well-dressed men are sitting together at a vacation resort.

'Farewell to Thee' is played in the background on Hawaiian guitar.

THIRD YORKSHIREMAN: But you know, we were happy in those days, though we were poor.

FIRST YORKSHIREMAN: Because we were poor. My old Dad used to say to me, "Money doesn't buy you happiness, son".

*and missing a lot of complaining in between, and maybe a
--little-- exaggeration*

FIRST YORKSHIREMAN: And you try and tell the young people of today that they won't believe you.

Of the sparkling wines, the most famous is Perth Pink. This is a bottle with a message in, and the message is 'beware'. This is not a wine for drinking, this is a wine for laying down and avoiding.

With thanks to the Monty Python team, especially Michael Palin