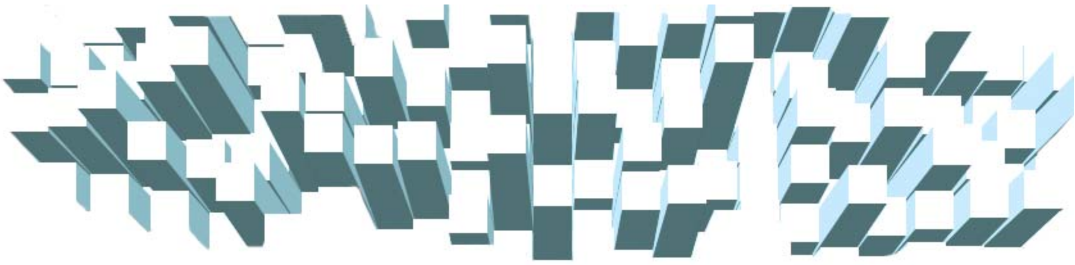


JPEG 2000 as a Preservation and Access Format for the Wellcome Trust Digital Library

Robert Buckley
Xerox Corporation

Edited by Simon Tanner
King's Digital Consultancy Services
King's College London

KDCS
Digital Consultancy

www.kdcs.kcl.ac.uk

Contents

© Buckley & Tanner, KCL 2009

1. Summary of Issues/questions

The Wellcome Trust is developing a digital library over the next 5 years, anticipating a storage requirement for up to 30 million images. The Wellcome previously has used uncompressed TIFF image files as their archival storage image format. However, the storage requirement for many millions of images suggests that a better compromise is needed between the costs of secure long-term digital storage and the image standards used. It is expected that by using JPEG2000, total storage requirements will be kept at a value that represents an acceptable compromise between economic storage and image quality. Ideally, JPEG2000 could serve as both a preservation format and as an access or production format in a write-once-read-many type environment.

JPEG2000 was chosen as an image preservation format due to its small size and because it offers intelligent compression for preservation and intelligent decompression for access. If a lossy format is used to obtain a relatively high compression, e.g. between 5:1 and 20:1 (in comparison to an uncompressed TIFF file), then the storage requirements desired are achievable. The questions to address are what level of compression is acceptable and delivers the desired balance of image quality and reduced storage footprint.

With regard to the use of JPEG 2000, the questions posed in the brief and addressed in this report are:

a. What JPEG2000 format(s) is best suited for preservation?

b. What JPEG2000 format(s) is best suited for access?

c. Can any single JPEG2000 format adequately serve both preservation and access?

d. What models exist for the use of descriptive and/or administrative metadata with JPEG2000?

e. If a JPEG2000 format is recommended for access purposes, what tools can be used to display/manipulate/manage it and any associated or embedded metadata?

This report will describe how a unified approach can enable JPEG2000 to serve for both preservation and access and balance the needs for compressed image size, image quality and decompression performance.

2. Recommendations

The majority of materials that will form part of the Wellcome Digital Library are expected to use visually lossless JPEG 2000 compression. Although “visually lossless” compression is lossy, the differences it introduces between the original and the image reconstructed from a compressed version of it are either not noticeable or insignificant and do not interfere with the application and usefulness of the image. Because the original cannot be reconstructed from the compressed image, compression in this case is irreversible. JPEG compression in digital still cameras is a familiar example of irreversible but visually lossless image compression. In mass digitization projects that use JPEG2000, compression ratios around 40:1 have been used for basically textual content. When applied to printed books, it has been found that these compression ratios do not impair the legibility or OCR accuracy of the text.

However, archiving and long-term preservation indicate a more conservative approach to compression and a different trade-off between compressed image size and image quality to meet current and anticipated uses. Still, given the volumes of material being digitized, a lossy format represents an acceptable compromise between quality and economic storage. A compression ratio of 4:1 or 5:1 gives a conservative upper limit on file size and decompressed image quality in the preservation format for the material being digitized. However this material should tolerate higher compression ratios with the results remaining visually lossless.

While most of the materials will use visually lossless compression, it is suggested that a small subset of materials (less than 5% of total) may be candidates for lossless or reversible compression. Reversible means that the original can be reconstructed exactly from the compressed image, i.e. the compression process is reversible.

Nevertheless, this report recommends irreversible JPEG2000 compression for the preservation and access formats of single grayscale or color images. Initially specifying a minimally lossy datastream will result in overall compression ratios around 4:1; the exact value will depend on image content. While this is a particularly conservative compression ratio, the compression can be increased as new materials are captured and even applied retroactively to files with previously captured material. The access format will be a subset of the preservation format with a subset of the resolution levels and quality layers in the preservation format.

In particular, the JPEG 2000 datastream should have the following properties:

- Irreversible compression using the 9-7 wavelet transform and ICT (see Section 3.1) with minimal loss (see Section 3.5)
- Multiple resolutions levels: the number depends on the original image size and the desired size of the smallest image derived from the JPEG 2000 datastream (see Section 3.2)
- Multiple quality layers, where all layers gives minimally lossy compression for preservation (see Sections 3.3 and 3.5)
- Resolution-major progression order (see Section 3.4)
- Tiles for improved codec (coder-decoder) performance, although the final decision regarding the use of tiles and precincts depends on the codec
- Generated using Bypass mode, which creates a compressed datastream that takes less time to compress and decompress (see Section 3.6)
- TLM markers (see Section 3.7)

A formal specification of the JPEG 2000 datastream for this application is given in Appendix 1. The datastream specified there is compatible with Part 1 of the

© Buckley & Tanner, KCL 2009

JPEG 2000 standard; none of the JPEG 2000 datastream extensions defined in Part 2 of the standard are needed.

Further, this report recommends embedding the JPEG 2000 datastream in a JP2 file. The JP2 file should contain:

- A single datastream containing a grayscale or color image whose content can be specified using the sRGB color space (or its grayscale or luminance-chrominance analogue) or a restricted ICC[1] profile, as defined in the JP2 file format specification in Part 1 of the JPEG 2000 standard (see Section 4.1)
- A Capture Resolution value (see Section 4.2)
- Embedded metadata that describes the JPEG 2000 datastream should follow the ANSI/NISO Z39.87-2006 standard and be placed in a XML box following the FileType box in a JP2 file (see Section 4.3)

Using the JP2 file format is sufficient as long as the requirement is for a single datastream whose color content can be specified using sRGB or a restricted ICC profile. While the JPX file format can be used if the color content of the image is specified by a non-sRGB color space or a general ICC profile, the use of a JP2-compatible file format is recommended.

3 Basis of recommendations/reasonings/tests done

3.1 Compression

In general terms, the compression ratio is set for preservation and quality, and JPEG 2000 datastream parameters such as the number of resolution levels and quality layers and tile size are set for access and performance. JPEG 2000 offers smart decompression, where only that portion of the datastream needed to satisfy the requested image view in terms of resolution, quality and location need be accessed and decompressed on demand and just in time.

The JPEG 2000 compression offers both reversible and irreversible compression. Reversible compression in JPEG 2000 uses the 5-3 integer wavelet transform and a reversible component transform (RCT). If no compressed data is discarded, then the original image data is recoverable from the compressed datastream created using these transforms. Irreversible compression uses the 9-7 floating point wavelet transform and an irreversible component transform (ICT), both of which have round-off errors so that the original image data is not recoverable from the compressed datastream, even when no compressed data is discarded. Appendix 2 contains a more detailed discussion of the differences between reversible and irreversible compression in JPEG 2000.

3.2 Multiple resolution levels

To begin with, it is recommended that JPEG 2000 be used with multiple resolution levels. The first two or three resolution levels facilitate compression; levels beyond that give little more compression but are added so that decompressing just the lowest resolution sub-image in the JPEG 2000 datastream gives a thumbnail of a desired size. For example, with a 5928-by-4872 pixel dimension original and 5 resolution levels, the smallest sub-image would have dimensions that would be 1/32 those of the original, in this case 186 by 153 pixels, which is roughly QQVGA[2] sized. Accordingly, JPEG 2000

[1] International Color Consortium, an organization that develops and promotes color management using the ICC profile format (www.color.org)

[2] Quarter-quarter VGA (Video Graphics Array); since VGA is 640 by 480, QQVGA is 160 by 120.

compression with 5 resolution levels is recommended for images of this and similar sizes, which are typical of the sample images provided. In practice, the number of resolution levels would vary with the original image size so that the lowest resolution sub-image has the desired dimensions.

3.3 Multiple quality layers

There are two main reasons for using multiple quality layers. One is so that it is possible to decompress fewer layers and therefore less compressed data when accessing lower resolution sub-images. This speeds up decompression without affecting quality since the incremental quality due to the discarded layers is not noticeable at reduced resolutions. The second reason is that multiple quality layers make it possible to deliver subsets of the compressed image corresponding to higher compression ratios, which may be acceptable in some applications. This means there is less data to transmit and process, which improves performance and reduces access times. It also means that it is possible for the access format to be a subset of the preservation format, derived from it by discarding quality layers as the application and quality requirements warrant. The use of quality layers makes it possible to retroactively reduce the storage needs should they be revised downward by discarding quality layers in the preservation format and turning images compressed at 4:1 or 5:1 for example into images compressed at 8:1 or higher, depending on where the quality layer boundaries are defined.

3.4 Example: TIFF to JP2 conversion

For example, the following command line uses the Kakadu[3] compress function (kdu_compress) to convert a TIFF image to a JP2 file that contains an irreversible JPEG 2000 datastream. In particular, it contains a lossy JPEG 2000 datastream with 5 resolution levels and 8 quality layers, corresponding to compression ratios of 4, 8, 16, 32, 64, 128, 256 and 512 to 1 for a 24-bit color image. These correspond to compressed bit rates of 6, 3, 1.5, 0.75, 0.375, 0.1875, 0.09375 and 0.046875 bits per pixel. (A compression ratio of 4 to 1 applied to a color image that originally had 24 bits per pixel means the compressed image will equivalently have a compressed bit rate of 6 bits per pixel.) The Kakadu command line use bit rates rather than compression ratios to specify the amount of compression.

```
kdu_compress -i in.tif -o out.jp2 -rate
6,3,1.5,0.75,0.375,0.1875,0.09375,0.046875 Creversible=no
Clevels=5 Stiles={1024,1024} Cblk={64,64} Corder=RPCL
```

The JPEG 2000 datastream created in this example has 1024-by-1024 tiles, 64-by-64 codeblocks and a resolution-major progressive order RPCL, so that the compressed data for the lowest resolution (and therefore smallest) sub-image occurs first in the datastream, followed by the compressed data needed to reconstruct the next lowest resolution sub-image and so on. This data ordering means that the data for a thumbnail image occurs in a contiguous block at the start of the datastream where it can be easily and speedily accessed. This data organization makes it possible to obtain a screen-resolution image quickly from a megabyte or gigiabyte sized image compressed using JPEG 2000. Tiles and codeblocks are used to partition the image for processing and make it possible to access portions of the datastream corresponding to sub-regions of the image.

[3] http://www.kakadusoftware.com/

3.5 Minimally Lossy Compression

The JPEG 2000 coder in this example would discard transformed and compressed data to obtain a compressed file size corresponding to 4:1 compression. This needs to be compared with the performance of the the minimally lossy coder, where no data is discarded but which is still lossy because of the use of the irreversible transforms. In some cases, depending on the image content, as shown in Section 3.6, the minimally lossy coder can give higher compression ratios than 4:1. Accordingly, it is recommended that a minimally lossy format with multiple quality layers and multiple resolution levels be used for the preservation format. The access format would use reduced quality subsets of the preservation format optionally obtained by discarding layers and using reduced resolution levels.

3.6 Testing reversible and irreversible compression

The reason to use irreversible compression is that it gives better compression than reversible compression, at the cost of introducing errors (or differences) in the reconstructed image. This section examines this performance tradeoff.

Reversible and irreversible compression were applied to four images provided by the Wellcome Digital Library (Figure 1). A variation on irreversible compression was tested which used coder bypass mode, in which the coder skipped the compression of some of the data. This gave a little less compression, but made the coder (and decoder) run about 20% faster. The Kakadu commands used in these tests are given in Appendix 2.

The compression ratios obtained with these three test are shown in the following table.

Original	Reversible	Irreversible	Irreversible w/bypass
L0051262_Manuscript_Page	2.25	3.45	3.42
L0051320_Line_Drawing	1.82	2.52	2.51
L0051761_Painting	2.46	3.96	3.90
L0051440_Archive_Collection	2.52	4.47	4.41

For these particular images, the compression ratio for irreversible JPEG 2000 was from about 40% to almost 80% better than it was for reversible, and on average over 30% faster (with a further 20% boost with coder bypass mode).

The cost of irreversible compared to reversible is the error it introduces. The error or difference between the reversibly and irreversibly compressed images is about 50 dB, which means the average absolute error value was about 0.5. For one of the sample images, 99.99% of the green component values were the same after decompression as they were before, or at most two counts different. (For the red and blue components, the percentages were 99.79 and 99.35.) This is within the tolerance for scanners: in other words, minimally lossless irreversible JPEG 2000 compression adds about as much noise to an image as a good scanner does.

A region was cropped from one image so that the visual effects of this error on this image could be examined more closely (Figure 2). When they were, the differences were not perceptible on screen or on paper. Unless being able to reconstruct the original scan is a requirement, legal or otherwise, then irreversible compression is clearly advantaged over reversible compression.

3.7 Further compression findings

In these tests, the compressed file sizes (and compression ratios) were image dependent and varied with the image content. Images with less detail or variation than these samples would give even higher compression ratios.

An advantage of JPEG 2000 is that it lets the user set the compression ratio, or equivalently the compressed file size, to a specific target value, which the coder achieves by discarding compressed image data. While this feature was not used to set the overall compressed file size in the minimally lossy compression case, it can be used to set the sizes of intermediate images corresponding to the different quality layers. The following Kakadu command line generates a JP2 file with a minimally lossy irreversible JPEG 2000 datastream that complies with the recommendation in this report:

```
kdu_compress -i in.tif -o out.jp2 -rate -, 4, 2.34, 1.36, 0.797,
0.466, 0.272, 0.159, 0.0929, 0.0543, 0.0317, 0.0185
Creversible=no Clevels=5 Stiles={1024,1024} Cblk={64,64}
Corder=RPCL Cmodes=BYPASS
```

The JPEG 2000 datatream in this example has 5 resolution levels and 12 quality layers. Using all 12 layers give a decompressed image with minimal loss. The intermediate layers boundaries are at pre-set compressed bit rates, starting at 4 bits per pixel, corresponding to a compression ratio of 6:1, assuming a 24-bit color original. Thereafter, the layer boundaries are distributed logarithmically up to a compression ratio of 1296:1. The exact values are not critical. What is important is the range of values and there being sufficient values to provide an adequate sampling within the range.

When a datastream has multiple quality layers, it is possible to truncate it at points corresponding to the layer boundaries and obtain derivative datastreams that correspond to higher compression ratios (or lower compressed bit rates). In the previous example, discarding the topmost quality layer produces a datastream corresponding to a compression ratio of 6:1 (compressed bit rate of 4 bits per pixel). Discarding the next layers produces a datastream with a compression ratio of 10.3:1, and so on. As noted previously, some images may have minimally lossy compression ratio greater than 6:1; the layer settings can be adjusted when this happens.

Using layers adds overhead that increases the size of the datastream and therefore decreases the compression ratio. To assess this effect as well as the overhead due to the use of tiles, the four sample images were compressed with one layer and no tiles, with one layer and 1024x1024 tiles, and with 12 layers and no tiles. As the following table shows, adding layers and tiles did decrease the minimally lossy compression ratio, but the effect was only visible in the third place after the decimal and was therefore judged insignificant in comparison to the advantages of using them.

Original	No tiles 1 layer	1024x1024 tiles 1 layer	No tiles 12 layers
L0051262_Manuscript_Page	3.452	3.450	3.443
L0051320_Line_Drawing	2.522	2.521	2.517
L0051761_Painting	3.961	3.957	3.948
L0051440_Archive_Collection	4.477	4.473	4.461

Besides tiles and layers, other datastream components that can improve performance and access within the datastream are markers, such as Tile Length Markers (TLM) which can aid in searching for tile boundaries in a datastream. Their effectiveness depends on whether or not the decoder or access protocol makes use of them. As a result, recommendations regarding their use depend on the choice of codec.

4 Implementation solutions / discussion

This section discusses the file format and metadata recommendations.

One function of a file format is packaging the datastream with metadata that can be used to render, interpret and describe the image in the file. Besides defining the JPEG 2000 datastream and core decoder, Part 1 of the JPEG 2000 standard also defines the JP2 file format which applications may use to encapsulate a JPEG 2000 datastream. A minimal JP2 file consists of four structures or “boxes”:

1. JPEG 2000 Signature Box, which identifies the file as a member of the JPEG 2000 file format family
2. File Type Box, which identifies which member of the family it is, the version number and the members of the family it is compatible with
3. JP2 Header Box, which contains image parameters such as resolution and color specification needed for rendering the image
4. Contiguous Codestream Box, which contains the JPEG 2000 datastream

4.1 Color Specification

How an image was captured or created determines the parameters in the JP2 Header Box, which are subsequently used to render and interpret the image. Among these parameters are the number of components (i.e. whether the image is grayscale or color), an optional resolution value for capture or display, and the color specification. In general the color content of an image can be specified in one of two ways: directly using a named color space, such as sRGB, Adobe RGB 98 or CIELAB, or indirectly using an ICC profile.

The digitization process and the nature of the material being digitized, not the file format, drive the color specification requirements of the application. The issue for the file format is whether or not it supports the color encoding used by the digital materials. What’s significant about the JP2 file format is that it supports a limited set of color specifications. For example, the only color space it supports directly is sRGB, including its grayscale and luminance-chrominance analogues. This is a consequence of the JP2 file format having been originally designed with digital cameras in mind.

Besides sRGB, the JP2 file format supports a restricted set of ICC profiles, namely gamma-matrix-style ICC profiles. This style of profile can represent the data encoded by RGB color spaces other than sRGB. The image data is still RGB; it’s just that it is specified indirectly by means of an ICC profile. This does not necessarily mean that non-sRGB systems must support ICC workflows; it does mean more sophisticated handling of the color specification in the JP2 file. For example, the system may recognize that the JP2 file contains the ICC profile for Adobe RGB 98 and use an Adobe RGB 98 workflow.

An alternative to JP2 is the Baseline JPX file format, defined in Part 2 of the JPEG 2000 standard. JPX is an extended version of JP2 which, among other things, specifies additional named color spaces, including Adobe RGB 98 and ProPhoto RGB. There are some RGB spaces, such as eciRGBv2, which JPX does not support directly and for which ICC profiles would still be needed for them to be used. The best thing is to use the JP2 file format as long as possible, since it is more widely supported than JPX and its use avoids support for the more advanced features of JPX when only extended color space support is desired.

4.2 Capture Resolution

The JP2 Header Box may also contain a capture or display resolutions, indicating the resolution at which the image was captured or the resolution at which it should be displayed. While the JP2 file is required to contain a color specification, it is not required to have either resolution values. Instead, it is up to the application to require it. This report recommends that the JP2 Header Box in the JP2 file contain a capture resolution value, indicating the resolution at which the image contained in the file was scanned. The JP2 file format specification requires that the resolution value be given in pixels per meter.

4.3 Metadata

In addition to the four boxes that a JP2 is required to contain, it may optionally contain XML and UUID boxes. Each can contain vendor or application specific information, encoded in an XML box using XML or in a UUID box in a way that is interpreted according to the UUID code (UUID stands for Universally Unique Identifier). These two types of boxes are used to embed metadata in a JP2 file. For example, UUID boxes are used for IPTC[4] or EXIF[5] metadata. An XML box can be used for any XML-encoded data, such as MIX.

While the application and system normally determine the nature and format of the metadata associated with an image, JPEG 2000-specific administrative or technical metadata is within scope for this report. While such metadata may or may not be embedded in a JP2 file, this reports recommends that it be embedded.

JPEG 2000-specific metadata in the JP2 file should follow the ANSI/NISO Z39.87-2006 standard. This standard defines a data dictionary with technical metadata for digital still images. It lists “image/jp2” as an example of a *formatName* value and lists “JPEG2000 Lossy” and “JPEG2000 Lossless” as *compressionScheme* values. Files that implement this recommendation would have “JPEG 2000 Lossy” as their *compressionScheme* value and would also contain a rational *compressionRatio* value.

Compression	compressionScheme	JPEG2000 Lossy
	compressionRatio	<rational value>

While “JPEG2000 Lossy” is the *compressionScheme* value for all files that follow this recommendation and the *compressionRatio* value can be derived from file size and parameters in the JP2 Header Box, it is recommended that they be specified explicitly.

The Z39.87 standard also defines a *SpecialFormatCharacteristics* container to document attributes that are unique to a particular file format and datastream. In the case of JPEG 2000, this container has two sub-containers: one for *CodecCompliance* and the other for *EncodingOptions*. The elements in the *CodecCompliance* container identify by name and version the coder that created the datastream, the profile to which the datastream conforms (Part 1 of the JPEG 2000 standard defines codestream or datastream profiles), and the class of the decoder needed to decompress the image (Part 4 of the JPEG 2000 defines compliance classes). The elements in the *EncodingOptions* container give the size of the tiles, the number of quality layers and the number of resolution levels.

[4] International Press Telecommunications Council, creates standards for photo metadata (http://www.iptc.org/IPTC4XMP/)

[5] Exchangeable image file format, a standard file format with metadata tags for digital cameras (http://www.exif.org/)

© Buckley & Tanner, KCL 2009

The following table shows the hierarchy of *SpecialFormatCharacteristics* containers and elements for JPEG 2000; the column on the right shows the values these elements would have for the data stream generated by the example in Section 3.7.

JPEG2000	CodecCompliance	codec	Kakadu
		codecVersion	6.0
		codestreamProfile	1
		complianceClass	2
	EncodingOptions	tiles	1024x1024
		qualityLayers	12
		resolutionLevels	5

An XML schema for these and the other elements defined in the Z39.87 standard is available at **http://www.loc.gov/standards/mix**.

When metadata is embedded in a JP2 file, it would be convenient if it were near the beginning of the file where it could be found and read quickly. Any XML or UUID boxes containing metadata can immediately follow the JPEG 2000 Signature and FileType boxes, which must be the first two boxes in a JP2 file. This means the metadata can come before the JP2 Header box, which in turn must come before the Contiguous Codestream box. Therefore, the metadata-containing boxes can occur in a JP2 file before any of the image data to which their contents pertain.

4.3 Support

JPEG 2000 is supported by several popular image image editors, toolkits and viewers. Among them are Adobe Photoshop, Corel Paint Shop Pro, Irfanview, ER Viewer, Apple QuickTime and SDKs from Lead Technologies and Accusoft Pegasus. Aside from the viewers, all offer automated command lines and batch support.

Other sources of JPEG 2000 components and libraries are Kakadu, Luratech[6] and Aware[7]. This is not an exhaustive list. JP2 is not natively supported by Web browsers. In this regard, it is like PDF and TIFF and, like them, Web browser plug-ins are available, such as from Luratech and LizardTech[8].

A common approach for delivering online images from a JPEG 2000 server is to decode just as much of the JPEG 2000 image as is needed in terms of resolution, quality and position to create the requested view, and then convert the resulting image to JPEG at the server for delivery to a client browser. This avoids the need for a client side plug-in to view JPEG 2000. The National Digital Newspaper Program (NDNP) uses this approach; it offers 1.25 million newspapers pages, all stored as JPEG 2000, on its website at http://chroniclingamerica.loc.gov/. While NDNP uses a commercial JPEG 2000 server from Aware, other commercial servers as well as the Djakota[9] open-source JPEG 2000 image server are also available.

The choice between a JPEG 2000 client or a JPEG client with server-side JPEG2000-to-JPEG conversion is a system issue that is largely independent of the JPEG 2000 datastream and file format recommendation in this report.

[6] http://www.luratech.com/

[7] http://www.aware.com/imaging/jpeg2000.htm

[8] http://www.lizardtech.com/download/dl_options.php?page=plugins

[9] http://www.dlib.org/dlib/september08/chute/09chute.html

© Buckley & Tanner, KCL 2009

5 Conclusion

This report has described the use of JPEG 2000 as a preservation and access format for materials in the Wellcome Digital Library. In general terms, the compression ratio is set for preservation and quality, and JPEG 2000 datastream parameters such as the number of resolution levels and quality layers and tile size are set for access and performance.

This report recommends that the preservation format for single grayscale and color images be a JP2 file containing a minimally lossy irreversible JPEG 2000 datastream, typically with five resolution levels and multiple quality layers. To improve performance, especially decompression times on access, the datastream would be generated with tiles and in coder bypass mode.

One consequence of using a minimally lossy JPEG 2000 datastream is that the compressed file size will depend on image content, which will create some uncertainty in the overall storage requirements. The ability with JPEG 2000 to create a compressed image with a specified size will reduce this uncertainty, but replace it with some variability in the quality of the compressed images. The sample images could tolerate more than minimally lossy compression; how much more depends on the quality requirements of the Wellcome Library, which will depend on image content. Until these requirements are articulated and validated, and even after they are, using quality layers in the datastream will provide a range of compressed file sizes to satisfy future image quality-file size tradeoffs.

This report recommends that the access format be either the same as the preservation format or a subset of it obtained by discarding quality layers to create a smaller and more compressed file. Requests for a view of a particular portion of an image at a particular size would be satisfied in a just-in-time on-demand fashion by accessing only as many of the tiles (and codeblocks within a tile), resolution levels and quality layers as are needed to obtain the image data for the view.

© Buckley & Tanner, KCL 2009

Appendix 1: JPEG 2000 Datastream Parameters

This table specified the values for the main parameters of the JPEG 2000 datastream.

JPEG 2000 Datastream Parameters	
Parameter	**Value**
SIZ marker segment	
Profile	Rsiz=2 (Profile 1)
Image size	Same as scanned original
Tiles	1024 x 1024
Image and tile origin	XOsiz = YOsiz = XTOsiz = YTOsiz = 0
Number of components	Csiz = 1 (graysale) or 3 (color)
Bit depth	Determined by scan
Subsampling	XRsiz = YRsiz = 1
Marker Locations	
COD, COC, QCD, QCC	Main header only
COD/COC marker segments	
Progression Order	RPCL
Number of decomposition levels	N_L = 5
Number of layers	Multiple (see text)
Code-block size	xcb=ycb=6
Code-block style	SPcod, SPcoc = 0000 0001 (Coding Bypass)
Transformation	9-7 irreversible filter
Precinct size	Not explicitly specified

Appendix 2: Reversible and Irreversible Compression

This appendix describes the operation of a JPEG 2000 coder and the differences between reversible and irreversible compression. The first thing a JPEG 2000 coder does with an RGB color image is to apply a component transform which converts it to something better suited to compression by reducing the redundancy between the red, green and blue components in a color image. The next thing it does is apply multiple wavelet transforms, corresponding to the multiple resolution levels, again with the idea of making the image more suitable for compression by redistributing the energy in the image over different subbands or subimages. The step after that is an optional quantization, where image information is discarded on the premise that it will hardly be missed (if it's not overdone) and will further condition the image for compression. The next step is a coder, which takes advantage of the all the prep work that has gone on before and the resulting statistics of the transformed and quantized signal to use fewer bits to represent it; this is where the compression actually occurs. The coder doesn't discard any data and is reversible, although the steps leading up to it may not be. In a JPEG 2000 coder, there is one more step in which the compressed data is organized to define the quality layers boundaries and to support the resolution-major progressive order mentioned earlier.

For JPEG 2000 compression to be reversible, there can't be any quantization or round-off errors in the component and wavelet transforms. To avoid round-off errors, JPEG 2000 has a reversible transforms based on integer arithmetic. So for example, JPEG 2000 specifies a reversible wavelet transform, called the 5-3 transform because of the size of the filters it uses. JPEG 2000 also specifies an irreversible wavelet transform, the 9-7 transform, based on floating point operations. The 9-7 transform does a better job than the 5-3 transform of conditioning the image data and so gives better compression, but at the cost of being unable to recover the original data due to round-off errors in its calculations. Because of these round-off errors, the 9-7 transform is not reversible.

The following Kakadu commands were used to generate the compressed images for the tests reported in Section 3.6. The first command generates a reversible JPEG 2000 datastream; the second, an irreversible but minimally lossy datastream; and the third, an irreversible datastream using coder bypass.

```
kdu_compress -i in.tif -o out.jp2 -rate - Creversible=yes
Clevels=5 Stiles={1024,1024 Cblk={64,64}} Corder=RPCL
//reversible
```

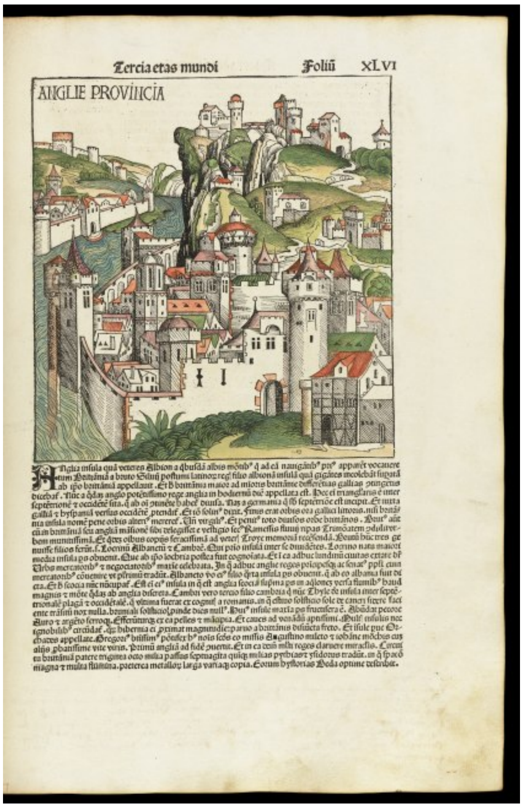
```
kdu_compress -i in.tif -o out.jp2 -rate - Creversible=no
Clevels=5 Stiles={1024,1024 Cblk={64,64}} Corder=RPCL
//irreversible
```

```
kdu_compress -i in.tif -o out.jp2 -rate - Creversible=no
Clevels=5 Stiles={1024,1024 Cblk={64,64}} Corder=RPCL
Cmodes=BYPASS
```

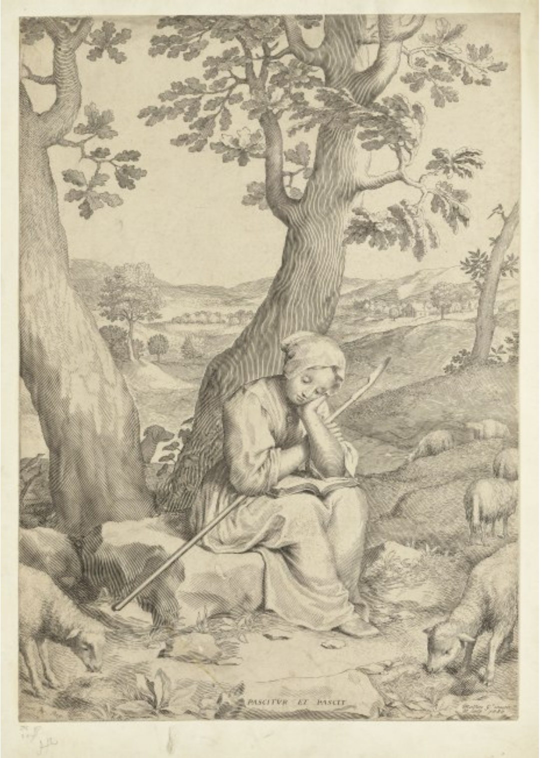
The dash after the rate parameter in these commands indicates that all transformed and quantized data is to be retained and none discarded. Creversible=yes in the first command directs the coder to use the reversible wavelet and components transforms. Creversible=no in the second and third commands directs the use of irreversible transforms. Cmodes=BYPASS in the third command directs the coder to use Bypass mode.

© Buckley & Tanner, KCL 2009

Figure 1: Sample images



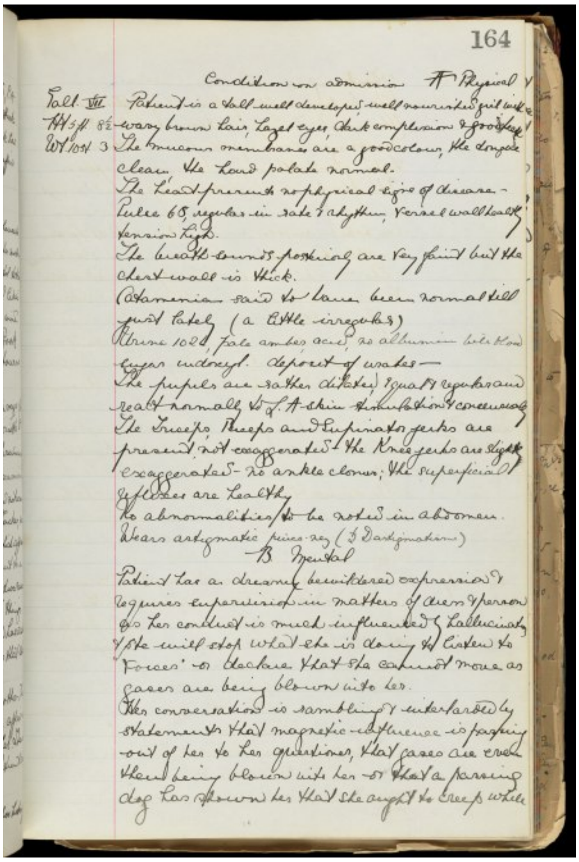
L0051262_Manuscript_Page



© Buckley & Tanner, KCL 2009

L0051320_Line_Drawing

L0051761_Painting



164

Condition on admission A Physical

Tall. VII Patient is a tall well developed well nourished girl with
Ht 5 ft 8½ wavy brown hair, hazel eyes, dark complexion & good teeth
Wt 10 st 3 The mucous membranes are a good colour, the tongue
clean the hard palate normal.
The heart presents no physical signs of disease -
Pulse 68, regular in rate & rhythm, vessel wall healthy
tension high.
The breath-sounds posteriorly are very faint but the
chest wall is thick.
Catamenia said to have been normal till
just lately (a little irregular)
Urine 1020, pale amber acid, no albumin bile blood
sugar indoxyl. deposit of urates—
The pupils are rather dilated, equal & regular and
react normally to L. A. skin stimulation & convergence
The Triceps, Biceps and Supinator jerks are
present, not exaggerated - the knee jerks are slightly
exaggerated - no ankle clonus; the superficial
reflexes are healthy
No abnormalities to be noted in abdomen.
Wears astigmatic pince-nez (½ D astigmatism)

B Mental

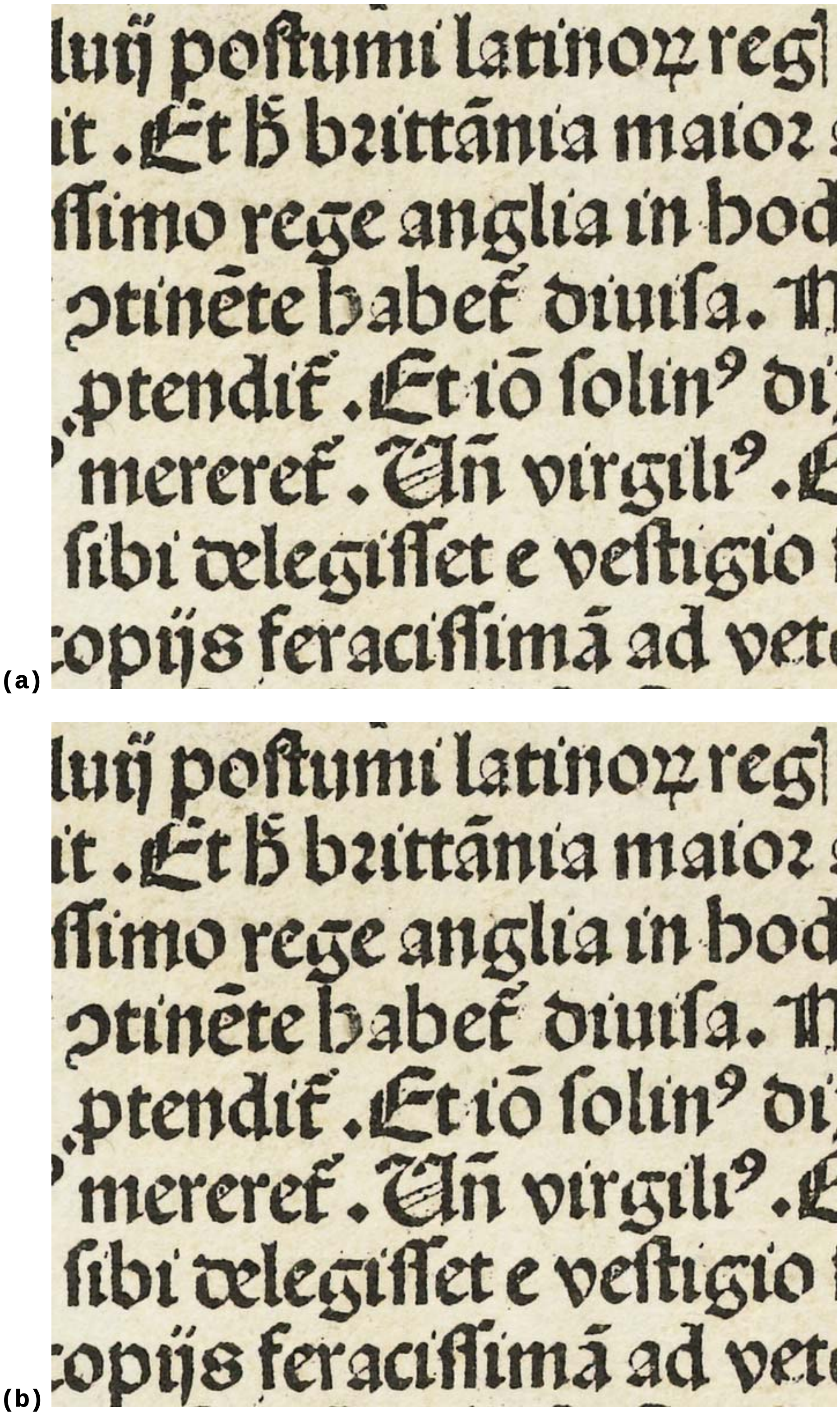
Patient has a dreamy bewildered expression &
requires supervision in matters of dress & person
as her conduct is much influenced by hallucinations
& she will stop what she is doing to listen to
'Forces' or declare that she cannot move as
gases are being blown into her.
Her conversation is rambling & interlarded by
statements that magnetic influence is passing
out of her to her questioner, that gases are even
then being blown into her - or that a passing
dog has shown her that she ought to creep while

L0051440_Archive_Collection

August 2009

© Buckley & Tanner, KCL 2009

Figure 2: Comparison of irreversible with reversible compression



luiȷ̄ poſtumi latinoꝝ reg
it .Et h̄ brittãnia maioꝛ
ſſimo rege anglia in hod
ꝯtinēte habet diuiſa. M
.ptendit .Et iō ſolinꝰ di
ꝰ mereret .Vñ virgiliꝰ .E
ſibi delegiſſet e veſtigio
copiȷ̄s feraciſſimã ad vet

luiȷ̄ poſtumi latinoꝝ reg
it .Et h̄ brittãnia maioꝛ
ſſimo rege anglia in hod
ꝯtinēte habet diuiſa. M
.ptendit .Et iō ſolinꝰ di
ꝰ mereret .Vñ virgiliꝰ .E
ſibi delegiſſet e veſtigio
copiȷ̄s feraciſſimã ad vet

Comparison of (a) irreversible compression with minimal loss and (b) reversible compression of region from L0051262_Manuscript_Page image, reproduced at 150 dpi.

© Buckley & Tanner, KCL 2009