Proc. SDIUT'01, Symposium on Document Image Understanding Technologies, Columbia MD, pp. 119–122, April 2001

Overview of the DjVu Document Compression Technology

Yann LeCun, Leon Bottou, Patrick Haffner, Jeffery Triggs
AT&T Labs - Research, Middletown, NJ
Bill Riemers, Luc Vincent
LizardTech, Inc., Seattle, WA
Contact: `lvincent@lizardtech.com`

Despite the growing importance of multimedia content, much of the knowledge, culture, and educational material in existence today is still available only in paper form. Bringing this wealth of information into the digital realm in a form that is faithful to the original, easily accessible, and searchable, is an essential step towards making the Internet the World's Universal Library.

DjVu (pronounced "deja vu") provides a way to do all this, and more. It was developed at AT&T Labs over the past several years and purchased by LizardTech in early 2000. DjVu is a compression technique, a file format, and a delivery platform that is specifically designed to enable the creation of digital libraries of printed documents (scanned from paper or digitally produced). It relies on a number of advanced content analysis techniques to achieve high compression ratios, low memory requirements, very fast rendering and indexing pf the material. These techniques have been thoroughly documented in the papers listed in the bibliography, and additional information is readily available on the web (`http://www.djvuzone.org and http://www.djvu.com` are good starting points).

A typical page from a book, magazine, or ancient document scanned in color at 300dpi contains on the order of 8 million pixels, and occupies 24MB uncompressed. Traditional compression techniques such as JPEG are notoriously inefficient on several counts:

- typical file sizes for a page will be between 400KB and 2MB at best, which is totally impractical for remote access.
- sharp edges (such as character outlines) are the cause of numerous wasted bits and/or unpleasant ringing artifacts.
- such large images are very slow to render, require a very large memory buffer for the decompressed image in the client, and are not easily zoomable or panable with current web browser technology.
- the text is not normally separated from the image, and therefore cannot be OCRed, indexed, or searched.
- no provision is made for multipage documents, unless one encapsulates the images into a container format such as PDF, thereby adding additional layers of inefficiencies.

The DjVu system alleviates these problems and can handle bitonal documents, low-color (palettized) images, photos and other continuous-tone images, scanned color or grayscale documents, as well as digitally produced documents (from PostScript or PDF).

Bitonal documents are encoded with a technique dubbed JB2, which builds a library of repeating shapes in the document (such as characters), and codes the locations where they appear on each page. Low-color images are compressed the same way, with the addition of a color palette, and a color index for each shape. Continuous-tone images are compressed with a progressive wavelet-based method dubbed IW44 that is on par with JPEG2000 in terms of signal to noise ratio, but whose decoder/renderer is very memory efficient, and extremely fast (3 times faster than the fastest JPEG-2000 mode).

Scanned color documents are decomposed into a foreground plane and a background plane. The foreground plane contains the text and the line drawings compressed as a bitonal or low-color image at maximum resolution (using JB2), thereby preserving the sharpness and readability of the text. The background plane contains the pictures and paper textures compressed at reduced resolution with IW44. Areas of the background covered by foreground components are smoothly interpolated so as to minimize their coding cost. The foreground/background segmenter first detects sharply contrasted areas, and then filters them with several criteria, such as their color uniformity, their geometry, and an estimation of their coding cost.

Digitally produced PDF or PostScript documents are turned into a list of low-level drawing commands using the popular tool GhostScript. This list is then translated into a list of non-overlapping shapes which are subsequently classified into the foreground or the background layer using a number of heuristics. The layers are then compressed as with scanned documents.

Bitonal documents in DjVu typically occupy 5 to 30KB per page at 300dpi, which is 3 to 8 times smaller than Group 4 (used in Fax machines, in TIFF files, and in PDF files). Low-color images such as icons are typically 2 times smaller than with GIF, but can be up to 10 times smaller if they contain lots of text. Photos are about 2 times smaller than JPEG, and about the same as fast modes of JPEG-2000 for the same SNR. An interesting aspect of IW44 wavelet codec is that it is optimized to allow on-the-fly decompression/rendering of the area visible in the display window (and not more) as the user zooms and pans around. This allows to keep the images in compressed form in the RAM of the client machine, and allows to display very large images without excessive memory requirements. Scanned color and grayscale documents in DjVu are typically 30 to 100KB per page at 300dpi, which is 5 to 10 times smaller than JPEG, and about 2-3 times smaller than MRC/T.44 or TIFF/FX. Digitally produced documents with mostly text are typically 2 times smaller than PDF or gzipped PostScript originals at 300dpi, but can be considerably smaller if the documents contain pictures.

DjVu documents are displayed within web browsers through a very compact plug-in (available for all major platforms). Everything in the design of DjVu was optimized to reduce the delay between the user's decision to view a page, and the display of that page on the screen. A multithreaded software architecture with smart caching allows individual document components to be loaded and pre-decoded on-demand. Pages are loaded on demand, allowing random access without prior download of the entire document, and without the help of a byte server. Page components (foreground layer, background chunks,...) are downloaded in sequence and rendered by a separate thread as soon as they are complete. This allows progressive rendering and refinement of the images. The page that follows the page currently being displayed is pre-loaded, pre-decoded and cached automatically thereby reducing the page-flipping delay. The DjVu viewer has a "modeless" graphical user interface that allows fast zooming, panning, and page flipping with a single mouse operation or keystroke.

The foreground layer can be OCRed and the result embedded back into the DjVu file as a searchable "hidden text" layer. Tools are available to extract that text and translate it into an XML format that includes each word, together with its bounding box coordinates on the page, and the document structure (pages, columns, paragraphs, lines, words). Hyperlinks, annotations, page thumbnails, and other metadata can also be embedded into DjVu documents.

Server-side full-text search can easily be provided using free indexing tools and a few Perl scripts. Large collections have been or are being put on the Web in DjVu with full-text search capabilities, including the NIPS Proceedings (13 volumes, 14,000 pages at 400dpi, 191MB), the Century Dictionnary (8 volumes), along with several national library collections and content from commercial providers around the world. DjVu is currently used by thousands of users to publish and exchange scanned documents on the Web.

DjVu can be seen as a general open platform for document delivery. Much of the code including the full IW44 codec, the palettized image compressor, and the multithreaded decoder/renderer (but not including the best segmenter and the best bitonal compressor) is available as open source under the General Public License (GPL) and can be used as a platform for research on new codecs, segmentation schemes, delivery mechanisms, viewing interfaces, and content analysis systems. More information, source code, benchmarks, and examples can be obtained at `http://www.djvuzone.org`. Plug-ins, compressors, and SDKs can be downloaded from `http://www.lizardtech.com`.

References

[1] L. Bottou, P. Haffner, P. Howard, P. Simard, Y. Bengio, and Y. LeCun. Browsing through high quality document images with DjVu. In *Proceedings of IEEE Conference on Advanced in Digital Libraries*, 1998.

[2] L. Bottou, P. Haffner, P. Howard, P. Simard, Y. Bengio, and Y. LeCun. High quality document image compression with DjVu. *Journal of Electronic Imaging*, 7(3):410–428, 1998.

[3] L. Bottou, P. Haffner, Y. LeCun, P. Howard, and P. Vincent. Un système de compression d'images pour la distribution réticulaire de documents numérisés (DjVu: An image compression system for distributing scanned document on the internet). In *Proceedings of CIFED, Conférence Internationale Francophone sur l'Ecrit et le Document*, Lyon, France, July 2000.

[4] L. Bottou, P. Howard, and Y. Bengio. The Z-coder adaptive binary coder. In *Proceedings of IEEE Data Compression Conference DCC'98*, pages 13–22, Snowbird, UT, Mar. 1998.

[5] L. Bottou and S. Pigeon. Lossy compression of partially masked still images. In *Proceedings of IEEE Data Compression Conference, DCC'98*, Snowbird, UT, Mar. 1998.

[6] P. Haffner, L. Bottou, P. Howard, and Y. LeCun. DjVu : Analyzing and compressing scanned documents for internet distribution. In *Proceedings of International Conference on Document Analysis and Recognition*, Bangalore, India, Sept. 1999.

[7] P. Haffner, Y. LeCun, L. Bottou, P. Howard, and P. Vincent. Color documents on the web with DjVu. In *Proceedings of IEEE International Conference on Image Processing*, Kobe, Japan, Oct. 1999.

[8] Y. LeCun, L. Bottou, , P. Haffner, and P. Howard. DjVu: a compression method for distributing scanned documents in color over the internet. In *Proceedings of Color 6, IST*, 1998.

[9] Y. LeCun, L. Bottou, A. Erofeev, P. Haffner, and B. Riemers. DjVu document browsing with on-demand loading and rendering of image components. In *Proceedings of SPIE, Internet Imaging II*, San Jose, CA, Feb. 2001.