Wide format raster compression applied to a printing environment

Cédric Sibade, Stéphane Barizien, Mohamed Akil, Laurent Perroton
Océ Print Logic Technologies, 1 rue J. Lemoine, 94015 Créteil Cedex, France
Lab. A^2SI, ESIEE Cité Descartes B.P. 99, 93162 Noisy-Le-Grand Cedex, France

Abstract

Digital compression allows to minimize the amount of data in a raster image. Its principle is to take advantage of the redundancy in its content. Typically it aims to decrease storage requirements and transfer times. In the wide format document industry, raster data are most of the time processed in an uncompressed format, in order to apply processing, color management or halftoning before printing. We anticipate the evolution of the data-flow size, by investigating the adequacy of compression, for wide format document management. Based on the image management and transformation features of the JPEG2000 compression standard, we propose to use a compressed format as the new data-flow representation.

1. Introduction

A paper document production chain has an equivalent structure to any other data processing chain: the input data is typically *acquired*, *processed* by different treatments, then *rendered*. The input or output media is either a paper or a digital data-flow. The acquisition is therefore either a scanning operation or a print submission, sent digitally through the driver's application. Likewise, the rendering is performed using a paper format –the so-called printing operation– or aims archiving, when a scan-to-file operation is asked.

As for any data processing structure, the results at any stage need transfers over communication layers, or temporary storage, using memory or disk media. For instance, spooling, print management or distant job submission may cause serious bottlenecks in terms of shared resource usage. Furthermore, the overall performance is degraded; processing time and data access are no longer optimized. Another drawback is the inherent cost increase, as expensive technologies are needed to handle this uncompressed format. This fact will become extremely limiting, when the paper document chain will focus on *wide format printing systems* (WFPS) or more generally wide format *document* systems (WFDS) to encompass products that include scanners. Not only the amount of data to process will drastically increase, but also huge stress will be put on the system.

In order to cope with these constraints, data compression techniques allow to (hopefully) represent these huge raster images with less data. Therefore the costly data transfer and storage space should decrease. The global resource use , i.e. memory and processor, should decrease; a smaller data format aims also to reduce the transfer times between the WFDS components.

In this paper, we anticipate the evolution of the data-flow size and propose the use of data compression as a new data-flow representation, a new encoding format taking care of the data redundancy. We will only focus on the raster type. Vectors primitives are finally sets of text symbols; they are therefore another type of *"compressed"* and high-level representation for geometric patterns. However no restrictions are put on the raster image characteristics used along this processing chain (any content, any color depth and any size).

In Section 2, we present the WFDS structure, with its generic components and its data-flow content. Section 3 reviews the compression features that have to be taken into account, as for wide format document management. With respect to these contributions of the compression, we propose in Section 4 to adopt a new representation for the data-flow based on the JPEG2000 compression. Conclusions and future works overview are finally provided in Section 5.

2. Wide Format Document Systems

2.1. Multi-Function Machine

Wide format products are no longer marketed as standalone machines. Scanners or printer engines belong to more modular systems, whose organization is centered on a core component. It is typically a very simple embedded controller that performs minimalist plot management functionalities. However, depending on the application and the desired functionalities a sophisticated RIP, for Raster Image Processor, can be used instead. One of the reasons of the development of these modular WFDS, is the possibility to build *Multi-Function Machines* (MFM). The following Figure 1 depicts a typical scanner-controller-printer MFM structure, with connections to a host application and a file server.

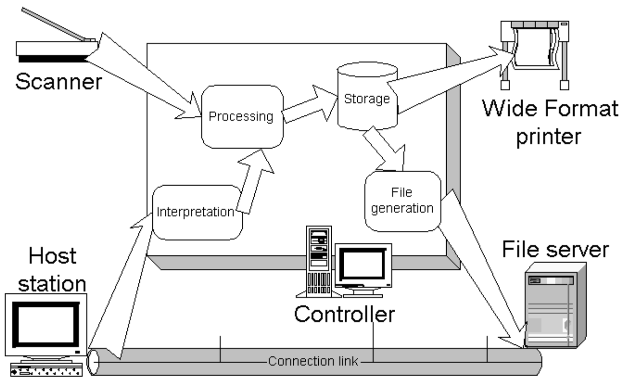

Figure 1: *Typical configuration of a MFM system.*

The central data controller's role is to manage and route the jobs from and to the different machines, and to perform the processing and the treatments that are not done by the other actors. Different functions are available using this very MFM structure: *printing* from the host application to the printer, *scan-to-file* from the scanner to the file server and *copying* from the scanner to the printer. RIPs extend this organization, focusing on the Graphic Arts market. Moreover, they offer a plethora of transformations, spooling and color management functions.

2.1.1. Components

The previous structure gathers only a limited number of generic components. Among the five main actors, scanners and host applications provide digital data. These *sources* feed different *processing* components, to handle geometric, color or management operations. *Storage devices* and *connections* are intermediate components to model the temporary backups and data transfers. The *sinks* such as the printer engine use digital inputs and produce paper outputs. Figure 2 illustrates a scanner-to-distant-printer chain with intermediate storage and different transforms.

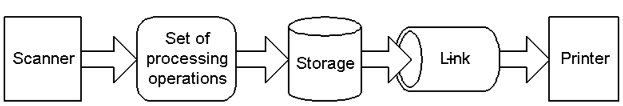

Figure 2: *Example of a copy chain*

These actors will have different behaviors and are characterized using generic properties, such as the input rate, their use of resource and their throughput.

2.1.2. Data-flow characterization

The other important characteristics of WFDS are their data-flows. The components, as described above, use and produce different kinds of data. The ”circulating” flows present some specific features, in relation with the transported contents. The pixel color depth (from 1 bpp for bi-level documents, to 24 bpp or more for ”true color”) and its density (or resolution expressed in pixels or dots per inch) varying from few to thousands of dpi greatly influence the resulting file size. Intuitively, this quantity increases in huge proportion as the document's physical dimension goes up to wide format paper. Digital wide format printing or scanning jobs can nowadays manage paper formats such as an A0 poster (46.81” by 33.11”); its equivalent file size in the case of a 24 bpp color coding and a resolution of 600 dpi is 1.6 GB. Moreover, the paper width is currently moving towards up to 60”. Most of the time, the engine will use roll-feed paper allowing banner prints or scans, extending the vertical size even longer.

2.2. Evolution of the content

The other feature that influences the data-flow characteristics and the processing operations is the document *content* (that is the ”meaning” of the carried data). Scanned or printed data can be of any type, from graphic or text raster (that is line-art data) to photos (also called continuous tone). By definition, it does not influence the overall size of the document, but has an effect on the processing operations (scaling, compression, segmentation…); to optimize the quality of these operations semantic-dependent algorithms may be used.

Furthermore, ready-to-print bitmaps present a third kind of contents. Grey-level or color documents are dithered to produce respectively 1 bpp or 4 bpp (one plane for each color, that is typically CMYK bit-planes as for the Cyan, Magenta, Yellow and Black colors).

Finally the trend for document contents is to mix different types of color or semantics: graphic arts posters or CAD maps now contain photos, computer-generated graphics and/or text. This evolution has also to be taken into account, as the processing algorithms that were tuned for one content will not produce an optimized performance for this mixed content raster.

2.3. Solutions for increasing rates

One of the first solutions to handle this ”size inflation” is to adapt the technology to handle these growing data: for instance faster links or network protocols are used and the memory or disk storage is increased. The counterpart of this proposition is the inherent cost increase. Even if the megabyte price will decrease in the long term and the processor speed or the network layers will propose better performance, this approach does not try to optimize the global cost of the final product.

This article proposes another solution, by looking at a *different representation* for the scanned or printed data: applying an adequate compression technique leads to smaller data flows. However, it also brings some limitations due to these transformed data.

3. Compression and Printing

Data compression presents a very attractive idea: an input document, with an input size flow N is reduced to a size, smaller than N, by reducing the data redundancy. The decompression process allows a (perfect or slightly modified) recovery of these input data. Applied to digital documents, the compression process aims at reducing the *data rate* (i.e. the transfer time is decreased) and the *storage* space (space saving). So speed is improved and cost decreased, as quality and service are not changed...

This simplistic and naive definition of compression is far from the reality. Compression/decompression is a process that uses resources (memory, CPU, transfer...) and does not preserve the quality of the input document (for lossy algorithms). However we will try to match the WFDS processing requirements with the compression issue.

3.1. Interesting Features

The main interesting feature of compression is its smaller representation. For lossy algorithms, a target bit-rate (at an optimized distortion value) can be established. If the content is consistent among all the input documents and adequate for lossless compression, the absence of expansion can be guaranteed. Current compression algorithms have been optimized in order to provide very impressive performances, for the resulting ratio and the quality of the reconstructed image.

Derived from the way they manage the input pixel data or developed to provide data access advantage, some compression techniques manage the data in an efficient way. The first-generation algorithms were treating each pixel independently or along a small neighborhood (looking for consecutive runs along lines or within short blocks). It implies that any data could not be accessed easily: to see and use one portion of the input raster, decompression should be applied to all its preceding image parts, following a fixed scanning route. Current approaches try to more decorrelate image parts by allowing to work on some sub-parts. JPEG [6] is using 8 by 8 blocks, JBIG [5] provides band splitting (horizontal stripes extracted from the image), the input raster is cut out by JBIG2 [2] into homogeneous content regions –of variable size– and finally JPEG2000 [3] allows to cut the image into rectangular sub-images which will be compressed independently. This tile or band splitting is a compulsory representation for wide format images, in order to lower the memory usage requirements of image processing operations.

Other interesting compression characteristics may be used for a printing production chain. Resolution progressivity (such as for JBIG or JPEG2000) provides access to the sub-sampled raster version of the input raster; these reduced representations of the input image can help for instance to build a viewer. For JPEG or JPEG2000, the spectral information (i.e. the spatial frequencies along the two dimensions) can be used for frequency-based processing operations. Finally, the pattern dictionary[1] created by the JBIG2 compression engine, targets applications using interpretation or character recognition.

3.2. A component of the chain

At first, compression seems to be limiting when placed along an image processing chain: this new data representation is no longer adequate for the other processing operations. A first approach to apply a rotation or a color transformation to a compressed stream first is to perform a decompression. The storage gain is lost as we need to recover an uncompressed flow; undesired decompression will furthermore add a processing cost to the overall treatment.

However, current compression algorithms describe methods to perform some of the usual geometric transformations directly on the compressed stream: squared-angle rotation and mirroring operations are easily applicable for the JPEG2000 standard. Extraction of sub-images is also available and scaling by a power of 2 is inherent from its resolution progressivity characteristic. Rotation and scaling ([1]) have also been developed for JPEG.

4. Proposed use of compression for WFDS

4.1. From optimized data-flows to full integration

The previously described benefits of the compression can be only applied to the "huge" WFPS targeted data-flows. In this first application of compression, the idea is to only optimize the problematic data-flows. If the constraints are put on the scanner-to-controller or the "host driver"-to-controller links, a more adequate content representation can be used. Likewise, the controller central storage can also benefit from one compression –smaller size, so smaller price–. The drawback of this kind of optimization is its restricted application. Only one flow will take advantage of the reduced format, and the compression and decompression stages surround the targeted links or storages.

[1]Each small connected component is a new entry in the dictionary. For other occurence of it, only the reference index is provided.

The extension of the previous approach is to adopt the compressed formats as a *new means of data-flow representation*. In an optimized version, the scanner and/or host driver produce a compressed file that will be easily transmitted and stored. Figure 3 illustrates this principle on the same example as figure 2.

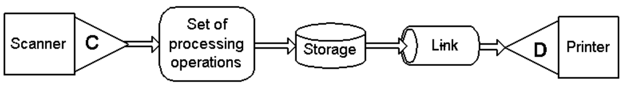

Figure 3: *Example of compression use for the copy chain. The compression* ***C*** *is applied as soon as the data are produced by the scanner. And in a fully optimized version, decompression* ***D*** *is performed at the last stage, before printing.*

Therefore the transformations have to be modified to take care of the new data format in order to avoid resource-consuming decompression (and re-compression if the compressed flow is needed for the continuing chain).

4.2. Compression choice

The new JPEG2000 standard was chosen ([4]). Not only the performance has reached an impressive level but also many other aspects were considered for our application. The configurable sub-image splitting is one of the answers to the wide format image use. From all the progressive modes that allow to order the compressed packets upon their spatial position, their color information, their quality or their resolution, the last one is useful in a WFDS. Either scaling or proof viewing will benefit from this resolution progressivity. As described above, 90° rotation or symmetry transformation is possible on the subband structure, basis for the compressed representation.

4.3. Multi-content issue

For the limitation introduced in the section 2.2 on the possibility to receive documents with different contents, JPEG-2000 will provide a format to efficiently handle this kind of image. It is based on a new paradigm called MRC, for Mixed Raster Content. Rather than processing the full image with one-content optimized algorithm, this raster is pre-separated into multiple layers (i.e. overlaid planes) with *homogeneous content*. Now that the content is stable over each layer, dedicated image processing operations or compressions can be applied on each of these planes.

This new concept is being adopted by other formats (like the DjVu or Algovision Luratech products), MRC-compatible file stream (with extension `.jpm`) to handle multi-layer rasters has been defined for JPEG2000. In that case, it is the role of the compression engine to adequately separate the input data and to "choose" the compression algorithms. As a first example, this MRC decomposition and encoding principle may be used for data-flows where the separation process has been implicitly done: the driver (which is the software application providing the flow to the controller, before printing) typically receives text or line primitives, separated from the pure raster parts. The two layers are therefore already existing.

5. Conclusion

JPEG2000 is a new raster representation. In addition to its interesting compression performance, it provides other interesting features, targeting image processing applications and for our problem, some of the WFDS requirements.

This use-case proposal of compression along a wide format document chain will be fully tested. A first step is to verify their adequation, by constructing a simulation of the processing chain; compression and decompression components can therefore be easily placed or removed to check the performance gain. Another study is strating on the transformations to be applied in the compressed domain, in order to extend the applicability range of the compressed format. Finally prototyping and integration will be carried on to demonstrate and market the solution.

Acknowledgements

This work is supported by the "Association Nationale de la Recherche Technique" (*ANRT*), under a CIFRE grant, to develop research collaborations between Océ Print Logic Technologies and the ESIEE A^{2}SI research laboratory.

References

[1] R. Dugad and N. Ahuja. A fast scheme for image size change in the compressed domain. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(4):461–474, April 2001.

[2] Paul G. Howard, Faouzi Kossentini, Bo Martins, Sren Forchhammer, William J. Rucklidge, and Fumitaka Ono. The emerging jbig2 standard. *IEEE Transaction on Circuit and systems for Video Technology*, 8, September 1998.

[3] M. Rabbani and R. Joshi. An overview of the JPEG2000 still image compression standard. *Signal Processing: Image Communication Journal*, 17(1):3–48, January 2002.

[4] Cédric Sibade, Stéphane Barizien, Mohamed Akil, and Laurent Perroton. Wide format image manipulation and compression in a printing environment. In *Proceedings of SPIE ITCOM 2002 conference on Multimedia Systems and Applications V*, August 2002.

[5] Stephen J. Urban. Review of standards for electronic imaging for facsimile systems. *Journal of Electronic Imaging*, 1(1):5–21, January 1992.

[6] Gregory K. Wallace. The JPEG still picture compression standard. *Comm. of the ACM*, 34(4):30–44, April 1991.