Implementing A Color Facsimile Machine

Giordano B. Beretta, Konstantinos Konstantinides, Daniel T. Lee, Ho John Lee and Andrew H. Mutz
Hewlett-Packard Laboratories, Palo Alto, California

Abstract

In 1994 the International Telecommunication Union – Telecommunications Sector has approved annexes to the Group 3 and Group 4 facsimile standards that allow for the exchange of continuous-tone color and gray-scale images. After reviewing the highlights of the new capabilities, we report on our experience in implementing this standard and testing the protocol. A discussion of the various components and the areas where we encountered challenges will help to identify those that need special attention for a successful implementation. We believe the quality of images transmitted with color facsimile is sufficient for many communication tasks of people working in the various color areas.

Background and Objective

Research and development of color facsimile can be dated prior to World War II.[7] This effort has resulted in expensive and proprietary systems for special applications, such as the transmission of color photographs by news delivery services. Three breakthroughs have occurred that will soon make color facsimile as ubiquitous as plain paper facsimile is now:

1. Electronic Imaging. New lossy data compression algorithms allow considerable reduction of the image data size in bytes. This size reduction makes it vi able to transmit images over the public-switched telephone network (PSTN) using ordinary modems.
2. Hardware Cost. High performance processors, color scanners, and color printers have become so economical that they can be assembled into a color facsimile machine with a purchase price accessible to departments or work groups.
3. International Standard. The International Telecommunication Union – Telecommunications Sector (ITU-T) has sanctioned a standard[4,5] in November 1994. This means that users can universally ex change color images with remote users without bothering about the manufacturer or type of the facsimile stations.

Two of the authors (D. Lee and A. Mutz) have been active in the standardization effort. In this paper we report on our implementation of the testbed that we used to verify the issues arising in the standardization process. As point 2 above suggests, the main issues for the design of a color facsimile machine are related to system integration and balancing. For this reason we believe that although we did not build an actual facsimile machine, our experience may be useful to build one.

The system design is not obvious in a time and age when commercial constraints require a richer set of features at higher performance and reduced cost. For example, at the baseline resolution of 200 dots per inch (dpi), an 8.5” × 11” (A size) binary image requires approximately 0.5M bytes of storage space, while a full color image requires 24 times as much, approximately 11M bytes. Conventional binary facsimile documents consist mostly of text, which can easily and efficiently be compressed because it consists mostly of white space and consecutive scan lines are highly correlated. The commercial constraint is to transmit a color document with pictorial contents in a similar time as with an existing facsimile machine, *i.e.*, for the same telephone calling cost.

In summary, we had two objectives: the facilitation of the standard and the implementation of a testbed. In the next section we will briefly present the features of T.30 facsimile protocol relevant to this paper. The following section will describe our implementation. A section on the results and lessons learned concludes the paper.

Facsimile Transmission

The new color facsimile communication provisions are true additions to the existing facsimile protocols, *i.e.,* color machines will be fully backwards compatible and fall back to the standard binary mode when communicating with older machines. Because the additions follow the same spirit for the Group 3 and Group 4 facsimile standards, we will limit our discussion to the Group 3 case. The Group 3 standard (ITU-T Recommendation T.30) is used over analog data channels and general public-switched telephone networks (PSTN). The companion Group 4 facsimile standard is used on public data networks, including packet-switched and integrated services digital networks (ISDN).

The T.30 recommendations used in Group 3 systems specify five time phases for facsimile transmission:

1. Phase *A* is where a *call* is *set up*. A connection is established, agreement is reached that both parties are facsimile machines, and the CSI (Called Station Identifier) is transmitted.
2. Phase *B* is known as the *pre-message procedure*. The called machine sends the DIS (Digital Identification Signal) at 300 bit/s with its capabilities. The caller replies with the DCS (Digital Command Signal) with the desired capabilities. Finally, the two machines negotiate (train) the transmission speed.
3. In phase *C* the actual *message transmission* occurs and a whole page is sent at the maximal speed viable on the actual phone line.
4. Phase *D* is known as the *post-message procedure*. The machines switch back to command mode (RTC,

Return To Control) and exchange an MCF (Message Confirmation).

5. Finally, in phase *E* the *call is released* with the trans mission of a DCN (Disconnect) signal.

The new additions for color affect only the pre-message and the message procedures. Because the changes in the pre-message procedure are a direct consequence of the changes in the message procedure, we have to discuss the latter first.

Message Procedure

The baseline coding for the binary mode T.30 is *Modified Huffman*. This compression method, as well as the enhanced variants *modified READ* and *modified modified READ*, are not well suited for gray-scale or full color images. For the new extension, baseline JPEG (Joint Photographic Experts Group)[1] has been adopted.

This method consists of three stages. First, a sequential discrete cosine transform (DCT) is applied, which is an orthogonal and separable transform that allows near-optimum energy compaction and for which a number of fast algorithms with low computational complexity have been developed.

In a second stage the data is quantized based on the discrete quantization table (DQT). This stage is lossy and implementors design the DQT so that no visible artifacts are introduced in the image. The compression ratio of a file can be increased by setting a so-called *q-factor* or *scaling factor*, which is essentially a uniform multiplicative parameter that is applied to the quantization tables.

The final step is a lossless Huffman encoder, which eliminates the entropy in the image file. The Huffman table (HT) controls the effectiveness of the lossless compression. At the receiving end, the inverse transforms are applied in reverse order, creating a rendition of the original image.

The color space is CIELAB with CIE illuminant D_{50}.[9] Particular attention has been given to the selection of an adequate gamut for the chrominance coordinates. Because the standard prescribes the allocation of 8 bits/pel/component (or optionally 12), a trade-off is necessary between the gamut size and the artifacts introduced by quantization into a limited data volume. It has been determined experimentally that with a gamut range of $a^* = [-80, 80]$, $b^* = [-80, 120]$, quantization artifacts under aggressive compression are relatively unobjectionable.[8]

The standard prescribes a default gamut range of $a^* = [-85, 85]$, $b^* = [-75, 125]$. For those cases in which *a priori* knowledge about the gamut range is available and high color fidelity[1] is necessary, the standard allows the specification of a custom gamut range. This custom gamut range can be changed on a *per* page basis.

The attributes of the image are encoded in the JPEG file in the form of APP1 application markers, so that the file is completely self-contained. These attributes include a unique FAX identifier, the spatial resolution, and optionally custom gamut range data. Unfortunately, no agreement could be reached on CIE illuminants other than D_{50} or on an arbitrary correlated color temperature.

Since the chrominance acuity of the human visual system is about half that of the luminance acuity, by default the color facsimile standards prescribes subsampling the image by 4:1:1 in the CIELAB space. For critical applications, it is possible to override this spatial subsampling in the chrominance channels.

Pre-message Procedure

Two groups of changes has been necessary in the pre-message procedure. The first is to change the baseline requirements and the second is to provide for the new capabilities.

As for the baseline requirements, they have been made more stringent for the error correction mode (ECM) and the resolution. Because a JPEG encoded image is a complex data structure, data integrity is essential. Therefore, ECM is a necessary prerequisite for color facsimile. The resolution must be at least 200 × 200 pels/25.4 mm because the JPEG method assumes square pixels, which is not the case in the lower facsimile resolutions.

The new capabilities are expressed by new bits in the DIS and DCS. This is an important aspect of the facsimile technology which ensures that its excellent user interface remains uncompromised. In the case of a mere file transfer, the sender would choose such attributes as the resolution, pass it along in the APP1 marker, and leave it to the receiver to figure out what resolution his equipment supports and how to resample the image.

In a facsimile system, the receiving machine will first communicate its capabilities in the DIS. Based on the sender's capabilities, the maximal subset is determined, communicated in the DCS, and used to create the JPEG encoded image. This ensures that the receiver can always render the image. Because normally the sender pays for the phone charges, the sending machine's capabilities—such as the resolution—can usually be restricted with buttons on the front panel.

Testbed Implementation

Equipment cost has not been an issue because our objective has been to assess the viability of a new standard, not the design of a product. Because of the profuse availability of public domain software components and the power of the software development tools available, we have implemented our testbed first on a Hewlett-Packard 9000 Series 700 workstation using the HP-UX operating system and the HP Visual User Environment (VUE). The peripherals are a ScanJet IIc scanner and a DeskJet 1200C printer.

Figure 1 shows a block diagram of the color facsimile pipeline. The process begins with the translation of a color, 24-bits per pixel, RGB image into the CIELAB color space. Next, the chroma components (a^*, b^*) of the image are subsampled by a factor of two in both the horizontal and vertical directions. The image is next coded using the baseline lossy JPEG algorithm. The compressed image is finally transmitted using the ECM mode of Group 3 facsimile protocol. On the receiver side, after Group 3 decoding, the image is JPEG decoded, color transformed, halftoned, and printed on a color or gray-scale printer.

We have implemented each block in Figure 1 as a separate ANSI C program and used the Unix pipe mechanism to implement the pipeline, except for the transmission part. To test the image processing pipeline stages, we pipe the JPEG encoder directly into the JPEG decoder. To test the Group 3 extensions to the facsimile protocol we have assembled a simple custom communications board based on the Rockwell R144EFX facsimile data pump and a line interface.

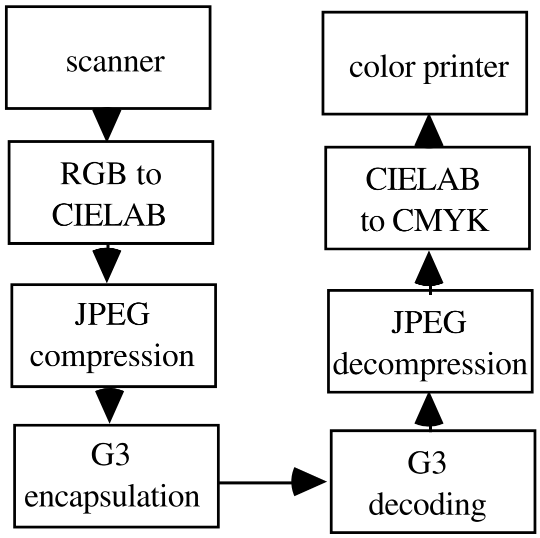

Figure 1. Block Diagram of the Color Facsimile Pipeline

Not shown in Figure 1 is the pipeline stage to handle the APP1 markers that are necessary for a facsimile JPEG data structure. Most JPEG encoder/decoder implementations ignore unknown application markers. For testing purposes, we have implemented a parser to read and manipulate the application markers in an image encoded with the JPEG method. With this parser we can also determine the type of a test file and set the appropriate bits in the DCS during the pre-message phase.

Once this Unix implementation has been fully operational and debugged, we have ported it to a Hewlett-Packard Vectra 486/66XM personal computer. The pipeline is executed in the MS-DOS environment. The peripherals are a Hewlett-Packard ScanJet IIc/ADF and a DeskJet 660C.

On the Vectra platform, we have implemented a pipeline mechanism mimicking Unix pipes.[6] The I/O streams are implemented as descriptors, so the data can be processed in-line and is never copied. Although the flat address space makes this easy to implement, only requiring a simple process monitor, the limits of a 16 bit architecture have been somewhat painful for an implementation of this size and complexity.

Experimental Results

Transmission Time

With our communications board we have been able to achieve error-free communication on switched external phone lines at speeds up to 9600 bit/s V.29, the limiting factor being the speed at which we can execute the modem functions in software. The effective facsimile transmission rate is approximately 64K bytes/min. This includes typical error correction mode overhead but not call establishment and negotiation times. Phases A and B require about 14 seconds.

Quantization

Using the default JPEG quantization tables, one can achieve perceptually lossless compression with compression ratios close to 20:1. For such a compression ratio and a transmission rate of 64K bytes/m, an A-size color image sampled at 200 dpi will require at least 8.6 minutes of transmission time. This is far longer than the generally known time for facsimile transmission. At higher compression ratios, acceptable printing quality (exceeding present expectations of facsimile quality) can be achieved. For example, for a transmission time of less than three minutes, the minimum required compression ratio is 60:1. If a modern V.34 facsimile modem is used at a speed of 28.8K bit/s, this compression ratio results in a very acceptable transmission time under 1 minute.

The question is how to achieve the 60:1 compression ratio. Even when the quantization tables are carefully designed to be perceptually lossless, a large *q*-factor will introduce artifacts, such as blockiness in areas of constant color or ringing on text characters. In an age when JPEG implementations have become integral part of operating systems (e.g., in Apple's MacOS), it often forgotten that the default JPEG quantization tables are just examples. In fact, according to information theory, the DQT elements should be chosen based on the energy within an 8×8 block.[2] More quantization levels are allocated to the frequency components with the larger variance. The statistical analysis of typical images transmitted via color facsimile has allowed us to achieve the desired compression rate.

System Overhead

Although the user of a facsimile machine is interested primarily in the transmission time, the time required to execute the various stages in the pipeline is also important, because it will dictate the required processing power and hence the purchase cost of the machine. For an A-size page sampled at 300 dpi we have measured the average timings shown in Table 1 on our Vectra implementation. The printing time includes dithering, generating PCL-3 code, and the actual printing time on the DeskJet 600C. The overhead for the pipeline code is included in the various stages where it is incurred.

Table 1. Typical execution times for the stages of a color facsimile testbed for images sampled at 300 dpi.

Pipe Stage	Time in Min:Sec
Scanning	1:55
Color mapping	2:19
Compression	1:18
Printing	2:33

Protocol Issues

Two technical points should be noted during the message phase. They relate to inter-line fills and the color space.

Inter-line Fills. The entire page must be encoded and transmitted strictly according to the JPEG standard and the error correction mode. In particular, it is not correct to introduce null octets into an ECM frame. This will cause a transmission failure due to corruption of the JPEG data structure. Introduction of null frames may occur because the conventional implementation practice is to require an inter-line fill across error-corrected block boundaries so that individual scan lines are not split between blocks.

Color Space. The second point is in the JPEG encoder/decoder. Once the image is decoded from DCT coefficients to spatial coordinates, a color space transformation is required before printing or display. Most color JPEG implementations have provisions for images represented in the 3-dimensional RGB and YUV color spaces. The YUV and the CIELAB color spaces are both luminance-chrominance spaces, yet YUV is quite different from the CIELAB space used for the color facsimile standard.

In addition, the CIELAB color space always requires the specification of a white achromatic stimulus. As noted earlier, the color facsimile standard assumes CIE standard illuminant D_{50}. If the colors are bluish or yellowish, the transformation from device coordinates to CIELAB or *vice versa* might have been carried out for an incorrect reference white or illuminant.

Conclusions and Outlook

We have outlined the new extensions to the standard ITU facsimile protocol that allow the transmission of gray-scale and full color images. We have verified the standard's viability by achieving a full implementation on a personal computer. For documents containing color images, the transmission time is approximately the same as on a conventional binary facsimile machine, however, at an incomparably superior image quality.

Following the standardization of hardcopy color facsimile, the standardization work continues on the soft-copy and palette-color extensions (using JBIG coding) to the color facsimile standard. Further, the problems and prospects for color management in the context of color facsimile need to be explored.

On the implementation side, we plan to explore the performance gains we might achieve by adding to the Vectra a custom board based on a commercially available, programmable, DSP (digital signal processor, a TMS320C50 from Texas Instruments).[6]

References

1. G. Beretta, "Experience with the new Color Facsimile Standard," *Proc. ISCC 64th Annual Meeting*, Greensboro, North Carolina, 28–36, April 1995.
2. W.-H. Chen and C. H. Smith, "Adaptive Coding of Monochrome and Color Images," *IEEE Transactions on Communications,* **COM-25**, 1285-1292, 1977.
3. IITU-T Rec. T.81 | ISO/IEC 10918-1 (JPEG), *Information technology—Digital compression and coding of continuous-tone images*, 1992.
4. ITU (Q.5/8 Rapporteur), *Amendments to ITU-T Rec. T.30 for enabling continuous-tone colour and grayscale mode for Group 3*, COM 8-43-E, June 1994.
5. ITU (Q.5/8 Rapporteur), *Amendments to ITU-T Rec. T.4 to enable continuous-tone colour and grayscale mode for Group 3*, COM 8-44-E, June 1994.
6. K. Konstantinides, H. J. Lee, G. Beretta, and A. H. Mutz, "A Testbed for Color Imaging and Compression," *Proc. Intern. Conf. on Digital Signal Processing*, Limassol, Cyprus, June 1995.
7. M. Matsuki and D. T. Lee, "Developments in Color-Facsimile Standards: An Overview," *Proc. SID 94 Digest*, San José, California, **XXV**, 883–886, June 1994.
8. A. H. Mutz, "Effect of Gamut Range Upon Color Quantization During JPEG Coding," *Proc. SID'94 Digest*, San José, California, **XXV**, 891–893, June 1994.
9. G. Wyszecki and W. S. Stiles, *Color Science*, Wiley & Sons, New York, 1982.

☆ This paper was previously published in *IS&T's 11th NIP Proc.,* p. 475 (1995).