# Disassembling of Composite Images

***Reem El Asaleh, Paul D. Fleming III and Alexandra Pekarovicova , Department of Paper Engineering, Chemical Engineering, and Imaging, Western Michigan University; Kalamazoo, MI***

## Abstract

*A method developed to exclude unwanted noise or marks from printed documents or images, where no original source of those images exists, is presented.*

*Using the fact that generally all digital images consist of a fixed set of rows and columns of pixels, where each pixel holds color information in terms of Red, Green and Blue colors, a C++ program was developed to disassemble a digital file in order to read and write its contents and reset its pixel colors. The investigated color images were scanned and digital files were obtained. This approach helped to clean an image or document from any unwanted overwritten marks. Finally, the new cleaned image was stored as another digital file***.**

## Introduction

A digital image can be thought as a digital or numerical representation of photographic paper or art, which can be created by scanning or by using a digital camera and stored in a computer for manipulation and enhancement procedures [1].

Generally, a digital image consists of a fixed set of rows and columns of pixels. These pixels (or picture elements) are the smallest construction units in the digital image and usually have a square shape. Each individual pixel holds numeric data that represent the brightness and the color of that image area in terms of red, green and blue [2].

The numbers of pixels in a digital image is referred as the resolution of the image [3]. Its also can be expressed as pixels per inch (ppi) or dots per inch (dpi). However, dpi is more accurately used to represent the resolution of a printer device, such as an inkjet printer [4]. Increasing the resolution will increase the total number of pixels covering the image area and it will also increase its quality and its storage file size. In addition, the overall size of an image is tied to its resolution, as the pixel size will increase if the resolution decreases [4]. The default resolution for a displayed image on a monitor will be 72 ppi, while for a good quality print, the image needs to have about 300 dpi resolution [2].

An image that is represented by a grid of pixels is also known as a raster graphic or a bitmapped graphic. Bitmapped graphics use the bits (binary digit), which are the basic unit of computer processing, to store the color information (i.e. 1 bit means the color could be white or black) [5]. The number of bits that is used to represent a color in a single pixel is called the bit depth (thus 1 bit-depth produce a binary image). For a grayscale image the 8-bits depth will generate an image with 256 shades of gray [6].

For colored digital images, each pixel holds color information in terms of red, green and blue and each color uses 8-bits to represent colors (or 1 byte). The result will be an RGB image with 24-bit depth (also known as true color) and with a total of 16,777,216 mixed colors [4].

The cones in the human eye are sensitive to light with particular wavelengths and, in particular, to the red, green and blue parts of the visible spectrum [7]. Additive and subtractive colors are two ways used to reproduce colors. Additive colors use the red, green and blue as primary colors. Secondary colors (cyan, magenta and yellow) can be produced by mixing any two of the primary colors. Mixing all the primary colors yields white. While the subtractive colors use the secondary colors as primaries. Mixing any two of the secondary colors will produce primary colors [8].

The CIE LAB, or CIE L*a*b*, and CIE LCH are two device independent ways used to describe color. The CIE (Commission Internationale de L'Éclairage) LAB color space is based on the human vision system, where the L* axis represents "Lightness" [L=0(black) – L=100 (white)]. The a* axis is green (represented by -a), and red at the other (+a). And the b* axis has blue at one end ( -b), and yellow (+b) at the other [9].

L*C*H* is presented in the form of a cylinder, where The L* axis represents Lightness and the C* axis represents Chroma or “saturation”, C=0 is completely unsaturated (i.e. a neutral grey, black or white) and C=100 maximum Chroma or saturation. The “Hue” is the circular axis (H°), where H=0° (red), H=90° (yellow), H=180° (green) and H=270° (blue). The two are not different color spaces, they are the same space, but one is in Cartesian coordinates (CIE LAB) and the other is in Cylindrical coordinates (CIE LCH) [9].

## Experimental

The goal of this research was to develop a method that can help cleaning the interrupting noise, or unwanted overwritten handwriting, from printed documents, where no original file exists.

For the purpose of this research and evaluating the quality of our method, one document was printed using two different digital printers (HP DeskJet F380 and HP LaserJet CP 3505).

Assuming that our document has no original digital file, the first step of this investigation was creating a digital version of these documents by scanning them using a high resolution (in this case we used 1200 dpi). The scanned files were then saved as BMP files.

Different scanned document samples were selected to test this method. These samples represent a variety of overwritten situations on text characters that include different pen ink and marker colors.

The main idea behind the cleaning method is to deal with the difference of ink color. Since the tested documents are now converted to digital files, rather than dealing with ink dot color, we are dealing with BMP pixel color.

The tested method depends on the LAB values of each pixel. Recalling the fact that each pixel holds color data in terms of RGB values, each pixel’s RGB value needs to be converted to its equivalent LAB value in order to test the cleaning method.

This method actually consists of two different approaches. The first approach is applying a chroma test that uses the absolute a* and b* values in the LAB color model, where any pixels having higher absolute values for a* and b* than a critical constant value (in this case 20) will be considered as colored pixels and will be set to white color (or cleaned). For black color (the text color), the a* and b* values are either equal to 0 or very low.

The second approach uses the L* (brightness) values where pixels that have high L* values indicate either a bright color or white (L*=100). Turning any pixels that have high L* value to white will leave the dark text color. However, instead of comparing each pixel with constant critical value, the mean and the standard deviation of all pixels's L* values were calculated first. Then each pixel's L* value was compared with the difference of the mean and the standard deviation. The mean value represents the average of the L* (brightness) values, while the standard deviation represents how much the brightness values vary. As a result of subtracting the mean from the standard deviation, a critical value will be generated, where all of the pixel's L* values that are higher than this number will be considered a very bright color and need to set to white (in other words cleaned). This approach improves the cleaning of the noise and improves the contrast of the text color as well.

To apply this method to the scanned document, a C++ program code was developed using Visual C++ 2008 software. This code will read the investigated BMP files; set their pixel colors and the new cleaned documents will be stored as other BMP format files.

## Results and Discussion

**Figure 1** shows a close look at one of BMP files that was generated from the document, printed on the HP DeskJet printer. This shows that each pixel has a different RGB value. The reason for that is when sending a digital image to be printed, its RGB values will be transformed to CMYK values which is equivalent to the printer device color model. Recall the fact that the same RGB value can be reproduced with different combinations of CMYK values. In addition, for the same printed document, the overall look could differ depending on the printing device, where each printing device employs a different printing process and ink. Also, it depends on the paper type that is used to print on.

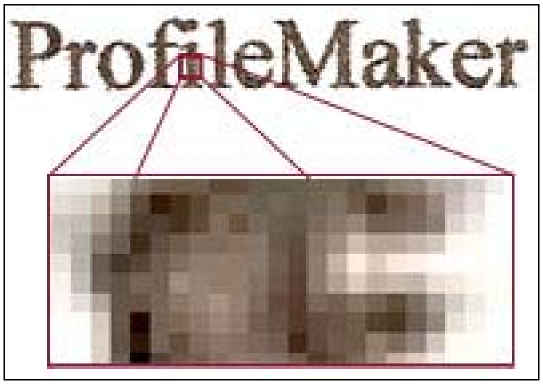


*FIGURE 1. Close looks of a BMP file showing the image pixel structure.*

**Figure 2** shows two different BMP files each representing different printing processes (i.e. LaserJet and DeskJet). Despite that each pixel has different RGB values; the overall pixels color appears, for the BMP file that represents LaserJet, darker than what represents the DeskJet. This allows more pixels that do not require any cleaning to maintain their color values after applying the cleaning method as demonstrated in **Figure 3**.

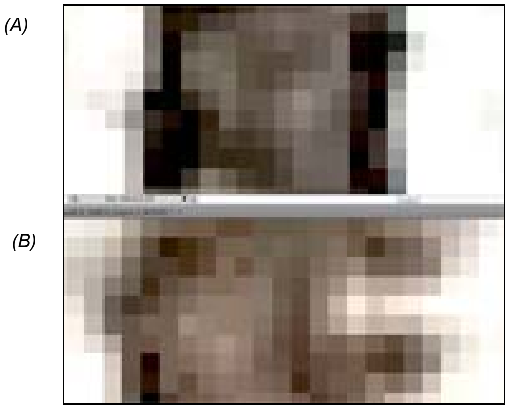


***FIGURE 2. A close look of two tested BMP files represent two printing process,*** *LaserJet (A) and DeskJet (B). Before cleaning process*

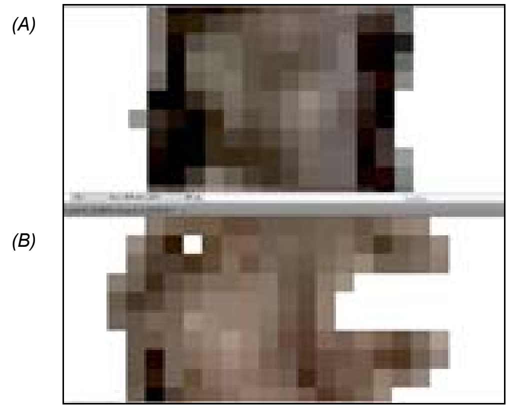
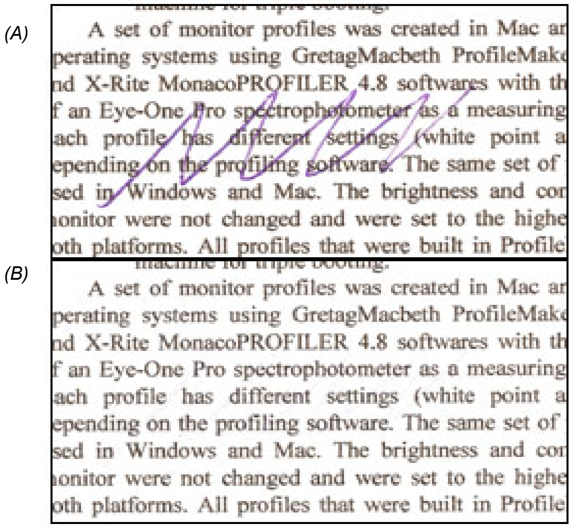


***FIGURE 3. A close look of two tested BMP files represent two printing process,*** *LaserJet (A) and DeskJet (B). After cleaning process*

Despite that the final cleaned images look different when closely viewed as in **Figure 3**, this could not be noticeable for the human eye, due to the small size of the pixels that represent a high resolution (in this research the resolutions is 1200 ppi).

Overall results on all tested images were close. The text characters were cleaned from any overwritten ink and became recognizable.

In **Figure 4** part A, a blue pen color had overwritten some of the document characters. Despite the cleaning of the document characters, some darker spots of the blue ink were not set to white color and therefore are still seen in the cleaned image as shown in part B. This means that these dark color pixels didn't pass either the brightness or the chroma test.

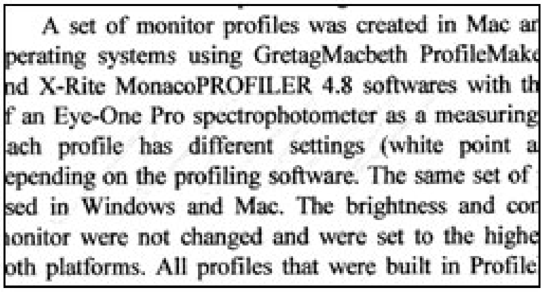


machine for triple booting.
A set of monitor profiles was created in Mac ar
perating systems using GretagMacbeth ProfileMak
nd X-Rite MonacoPROFILER 4.8 softwares with th
f an Eye-One Pro spectrophotometer as a measuring
ach profile has different settings (white point a
epending on the profiling software. The same set of
sed in Windows and Mac. The brightness and cor
nonitor were not changed and were set to the highe
oth platforms. All profiles that were built in Profile

machine for triple booting.
A set of monitor profiles was created in Mac ar
perating systems using GretagMacbeth ProfileMak
nd X-Rite MonacoPROFILER 4.8 softwares with th
f an Eye-One Pro spectrophotometer as a measuring
ach profile has different settings (white point a
epending on the profiling software. The same set of
sed in Windows and Mac. The brightness and cor
nonitor were not changed and were set to the highe
oth platforms. All profiles that were built in Profile

***FIGURE 4. A scanned document with a blue ink pen*** *(was printed using the HP DeskJet printer) before (A) and after (B) the cleaning process.*

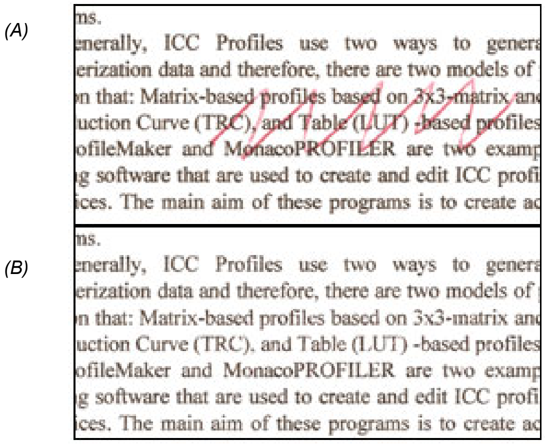
Also, each cleaned pixel was set to white color, while the other remaining pixels didn't change in color. However, in **Figure 5** all the untouched pixels were set to black color which gave better contrast for the text characters.



A set of monitor profiles was created in Mac ar
perating systems using GretagMacbeth ProfileMak
nd X-Rite MonacoPROFILER 4.8 softwares with th
f an Eye-One Pro spectrophotometer as a measuring
ach profile has different settings (white point a
epending on the profiling software. The same set of
sed in Windows and Mac. The brightness and cor
nonitor were not changed and were set to the highe
oth platforms. All profiles that were built in Profile

***FIGURE 5.*** *Part B from Figure 4 the BMP pixels were set to black color.*

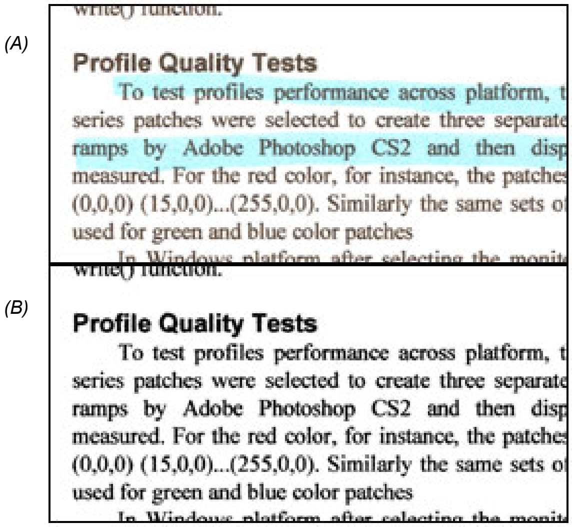
The red ink from **Figure 6** part A was totally cleaned and all the colored pixels pass the brightness and the chroma test, therefore there are no signs of noise in the cleaned document. The resulting image is demonstrated in part B.



ns.
enerally, ICC Profiles use two ways to genera
erization data and therefore, there are two models of
n that: Matrix-based profiles based on 3x3-matrix an
uction Curve (TRC), and Table (LUT) -based profiles
ofileMaker and MonacoPROFILER are two examp
g software that are used to create and edit ICC profi
ices. The main aim of these programs is to create ac

ns.
enerally, ICC Profiles use two ways to genera
erization data and therefore, there are two models of
n that: Matrix-based profiles based on 3x3-matrix an
uction Curve (TRC), and Table (LUT) -based profiles
ofileMaker and MonacoPROFILER are two examp
g software that are used to create and edit ICC profi
ices. The main aim of these programs is to create ac

***FIGURE 6. A scanned document with red ink pen*** *(was printed using the HP DeskJet printer) before (A) and after (B) cleaning process*



write() function.

Profile Quality Tests

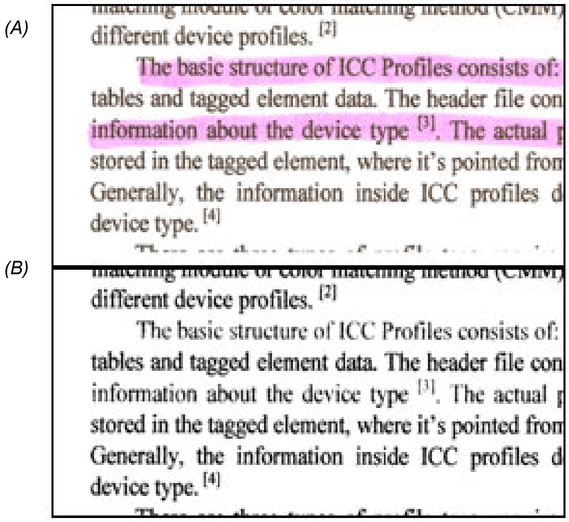
To test profiles performance across platform, t
series patches were selected to create three separate
ramps by Adobe Photoshop CS2 and then disp
measured. For the red color, for instance, the patches
(0,0,0) (15,0,0)...(255,0,0). Similarly the same sets o
used for green and blue color patches
In Windows platform after selecting the monit

write() function.

Profile Quality Tests

To test profiles performance across platform, t
series patches were selected to create three separate
ramps by Adobe Photoshop CS2 and then disp
measured. For the red color, for instance, the patches
(0,0,0) (15,0,0)...(255,0,0). Similarly the same sets o
used for green and blue color patches
In Windows platform after selecting the monit

***FIGURE7. A scanned document with a cyan marker*** *(was printed using HP DeskJet printer) before (A) and after (B) cleaning process*



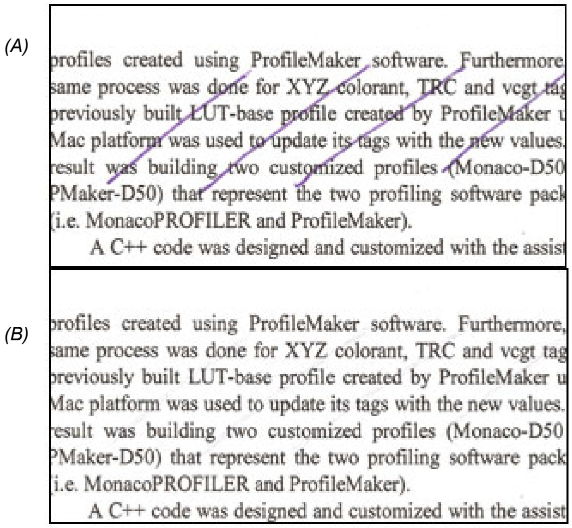
matching module or color matching method (CMM)
different device profiles. [2]
The basic structure of ICC Profiles consists of:
tables and tagged element data. The header file con
information about the device type [3]. The actual p
stored in the tagged element, where it's pointed from
Generally, the information inside ICC profiles d
device type. [4]

matching module or color matching method (CMM)
different device profiles. [2]
The basic structure of ICC Profiles consists of:
tables and tagged element data. The header file con
information about the device type [3]. The actual p
stored in the tagged element, where it's pointed from
Generally, the information inside ICC profiles d
device type. [4]

***FIGURE 8. A scanned document with a magenta marker*** *(was printed using HP LaserJet printer) before (A) and after (B) cleaning process*

**Figures 7 and 8** part A used two different marker colors (cyan and magenta). Both resulting images (part B) from the software showed cleaned document characters.

All the above BMP samples were taken from a document that was printed using an HP DeskJet printer. A second set of BMP samples were taken from the same document but this time it was printed using the HP LaserJet printer. The BMP files were processed through the same C++ software and the overall results were similar to the first set of BMP files (**Figure 9**). However, the only difference that all the unchanged pixels maintain their color values and did not reset them to black, due to the better quality printing of the LaserJet printer.



(A)

profiles created using ProfileMaker software. Furthermore
same process was done for XYZ colorant, TRC and vcgt tag
previously built LUT-base profile created by ProfileMaker u
Mac platform was used to update its tags with the new values.
result was building two customized profiles (Monaco-D50
PMaker-D50) that represent the two profiling software pack
(i.e. MonacoPROFILER and ProfileMaker).
A C++ code was designed and customized with the assist

(B)

profiles created using ProfileMaker software. Furthermore,
same process was done for XYZ colorant, TRC and vcgt tag
previously built LUT-base profile created by ProfileMaker u
Mac platform was used to update its tags with the new values.
esult was building two customized profiles (Monaco-D50
PMaker-D50) that represent the two profiling software pack
i.e. MonacoPROFILER and ProfileMaker).
A C++ code was designed and customized with the assist

***FIGURE 9. A scanned document with a blue ink pen*** *(was printed using the HP LaserJet printer) before (A) and after (B) cleaning process*

## Areas for Future Research

- Converting RGB values to CIE XYZ is accomplished using a 3x3 transform matrix along with reference white point. This software uses the standard sRGB white point. In case the scanned image has an embedded profile, it should use the profile and its reference white point for more accurate conversion.
- A new approach needs to be investigated to save the cleaned image as PSD (Photoshop) format. The advantage is to create separated layers for the background and foreground images and being able to combine these entire layers into a single file.

## Conclusions

Generally dark colors have low L* , a* and b* values, as a results dark color pixels might not pass the brightness test or the chroma test, which may still appear in the document file as unclean pixels or even appear around the text characters as well. These cases might be considered as limitations of the new approaches and more investigations need to be done to separate the dark color ink from the text characters.

## References

[1] Fulton, W. (2008), “What is a digital image anyway?”, see http://www.scantips.com/basics1b.html

[2] Lacey, J. (2001), *The Complete Guide to Digital Imaging*, Watson-Guptill, 1st Ed, pg 14

[3] Wikipedia (2009a), “Image resolution”, see http://en.wikipedia.org/wiki/Image_resolution

[4] Galer, M. and Horvat, L. (2005), *Digital imaging*, Elsevier, 3rd Ed , pg 2-16

[5] Wikipedia (2009b), “Raster graphics”, see http://en.wikipedia.org/wiki/Raster_graphics

[6] Umbaugh, S. (2005), *Computer imaging: digital image analysis and processing*, CRC Press, 1st Ed, pg 45

[7] Hardeberg, J. Y. (2001), *Acquisition and Reproduction of Color Images: Colorimetric and Multispectral Approaches*, Universal-Publishers, 2nd Ed, pg 10

[8] Sharma A. (2004), *Understanding Color Management*, Delmar Thomson Publishing, 1st Ed, pg 31

[9] Fairchild, M. D. (2006), *Color Appearance Models*, 2nd Ed, John Wiley & Sons Ltd, pg 79

## Author Biographies

***Reem El Asaleh*** *received her B.Sc. in Computer Science from UAE University in Al-Ain. She received her MS in Paper and Imaging Science and Engineering and is currently enrolled in the PhD program at Western Michigan University.*

***Paul D.“Dan” Fleming*** *is Professor in the Department of Paper Engineering, Chemical Engineering and Imaging at Western Michigan University. He has a Masters in Physics and a PhD in Chemical Physics from Harvard University. His research interests are in digital printing and imaging, color management and interactions of ink with substrates. He has over 250 publications and presentations and 1 US patent. He is a member of the IS&T, TAGA, TAPPI and the American Physical Society.*

***Alexandra Pekarovicova****, Ph.D., is an Associate Professor in the Department of Paper Engineering, Chemical Engineering and Imaging at Western Michigan University. She gained her Ph.D. in Chemical Engineering at Slovak Technical University. She has published over 40 peer-reviewed papers, and over 60 conference proceeding articles in ink and substrate interactions, and print quality.*