Detection of Monochrome Pages in Color Document Scans

Nathir A. Rawashdeh, Mohamed N. Ahmed, Lexmark International, Inc., Lexington, KY, USA

Abstract

In this paper, we present an algorithm for detecting monochrome pages in a color copy job on a multi function printer (MFP) with a contact image sensor (CIS) based scanner. Once detected, a monochrome page can be processed as such, which can improve image quality, print speed, and save color-printing supplies. The presented algorithm processes the RGB color data captured with the CIS scan bar for a given scan band and keeps track of color information across all scan bands moving down the page. The colorfulness of every pixel in the scan bands is derived from the Cb and Cr channels after color conversion from the RGB to the YCbCr space. A pixel is classified as color if its colorfulness value is greater than a predetermined device and media specific threshold. This threshold is found by modeling the cumulative colorfulness histogram of a number of scanned test documents using a mixture of two Gaussian distributions and the Expectation Maximization (EM) algorithm. For every scan band, the highest concentration of color pixels is saved, and later used to classify the page content as either color or monochrome.

Introduction

Multi function printers have become prevalent in recent years, because they are designed to offer high functionality at relatively small size and cost. Common MFP functionality includes print, scan, fax, and copy. The copy function involves a scan of the document, either on the flatbed glass or through the automatic document feeder (ADF), followed by printing the digitized document. In order to offer high speed at low cost, most MFP's employ contact image sensor based scanners. The CIS is an intensity sensing bar that typically spans the width of the document. As the scan bar moves down the document page, red, green, and blue light is shown on the portion of the document in front of the scan bar. This enables the CIS to obtain RGB intensity values of the scanned document – one color at a time. The data is then processed in an image processing pipeline [1], that prepares the document's RGB image for printing and alleviates scanner induced defects such as color fringing, aliasing, color noise, and blurring.

Color fringing occurs because the CIS scan bar does not stop as it strobes the red, green, and blue light to illuminate the page content. This causes the R, G, and B pixel values to not be representative of the exact content. Aliasing is caused by page content that is of higher spatial frequency than the scan resolution. High frequency content appears as lower frequency distortion in this case. An anti aliasing filter can mitigate this problem by removing the frequencies higher than the scanner can resolve. Color noise appears as random color variations in the scan and is due to inherent noise in the scan bar sensors, which causes differences between the color channels. A color neutral gray patch, for example, must be recorded as equal parts of red, green and blue signal – it will have hue otherwise. In general, noise in the scan bar is likely to cause unequal R, G, and B values for a gray pixel. Blurring is caused by imperfect lenses and optical elements, and is mitigated using sharpening filters. In addition to these hardware related defects, page content also requires special processing. MFP's usually offer different settings, or copy modes, to process the document content uniquely, such as photo, graphics and text. Figure 1 shows an enlarged region in a raw CIS RGB scan of an electrographic print including a magenta fill and numbers printed with black toner on plain white paper. The enlarged region illustrates scan defects such as color noise and color fringing. The isotropic color noise is evident in the flat areas of the solid magenta and paper white. This causes color neutral document content to appear slightly colored, i.e., have non-zero chroma. In addition, color fringing causes red and blue color traces along horizontal edges as can be seen around the black number characters in the figure. This also introduces color pixels into the scan even if the content is color neutral. Another source of colored pixels is the back side of the document, as colored content can show through the page from the back side, especially when the paper is thin [2].

This paper presents an algorithm for the detection of monochrome pages in a copy job on an MFP with a CIS based scanner. When a monochrome page is recognized, it can be printed as such, which saves color printing supplies, time, and energy. A similar objective has been presented by Dong *et al.* [3]. A color/monochrome page classifier has been implemented as part of an automatic copy mode selection heuristic. Their solution divides the scanned image into blocks with tunable size. The colorfulness of pixels in each block is computed as the Manhattan Distance from the neutral axis in the RGB color space. The colorfulness of a block is defined as the average colorfulness of the pixels contained within the block. Using a threshold for block colorfulness, a page is classified as color if at least one block was deemed colorful. On a higher level, Hirota *et al*. describe methods for classifying a page as either color or monochrome based on analyzing the proportions of color present in the image blocks [4]. Additionally, Handley shows how to apply the Expectation Maximization algorithm to classify pixels in color scans into either background or foreground pixels, based on processing the pixels in the CIE L*a*b* color space [5, 6, 7].

Our algorithm also divides the scanned image into blocks; however, the blocks are processed one scan band at a time to find the percentage of colorful pixels within the blocks. For a given scan band, only the maximum percentage of color amongst all blocks is saved for later processing when the last band is scanned. We define the colorfulness of a pixel as its distance to the neutral axis in the YCbCr opponent color space [8]. A colorfulness threshold is used to classify a pixel as color or monochrome. This threshold can be manually set, by trial and error, such that the pixels are properly classified given the specific device and media

at hand; however, our approach computes the cumulative colorfulness histogram of a number of representative color and monochrome test documents, and models it using the Expectation Maximization algorithm to find the best two Gaussian distributions that model the monochrome and color classes. The optimal threshold for colorfulness is then, the intersection of the two distributions. One advantage of this approach is that the cumulative colorfulness histogram model accuracy improves the more test documents are included in the analysis. After the data from the last scan band has been processed, the algorithm analyzes the maximum concentration of color pixels in each band's blocks. The result is a graph with peaks at the band locations that have a high color concentration.

The results show that this approach is robust to isolated color pixels from color noise or show-through, while still detecting small regions of concentrated color such as an isolated colored word or letter. The processing required to implement this solution as a copy mode feature is relatively small and includes analyzing each pixel only once for colorfulness, and counting each colored pixel once. In addition, the value of the maximum color concentration must be saved for each band and compared to a threshold. This simplicity enables its implementation in a low-end image pipeline.

***Figure 1**. An enlarged portion of a raw CIS scan of a laser print of a magenta fill and black numbers on white paper showing blur, color fringing, and noise.*

Color Processing

Typically, when a page is scanned, a scan bar moves down the page while illuminating the portion of the page in front of the scan bar sensor. The scanner collects the RGB intensities of pixels in the scan band and starts processing them immediately, before the scan bar reaches the end of the page. This approach is cost effective because the scan bar sensor is a relatively expensive element and can be kept small. In addition, the small scan bar sensor minimizes the amount of data that must be processed per unit of time. This reduces memory requirements, and simplifies the processing hardware.

The multi-function printer used in this experiment processes the scan band pixels in the YCbCr opponent color space because it encodes color information more efficiently than the RGB space [1, 5, 8]. The following equation is used to convert scan band pixels from the RGB to the YCbCr color space:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.2989 & 0.5866 & 0.1145 \\ -0.1687 & -0.3312 & 0.5000 \\ 0.5000 & -0.4183 & -0.0816 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix}, \qquad (1)$$

where *R*, *G*, *B*, *Y*, *Cb*, *Cr* are 8-bit values ranging from 0 to 255. The RGB values are not standardized and vary with the scanner model and hence the resulting YCbCr gamut is also scanner specific. The *Y* channel is equivalent to a grayscale (luma) image, and the color information is encoded in the *Cb* and *Cr* channels. The *Cb* and *Cr* channels encode the blue and red color content respectively. *Cb* and *Cr* values less than 128 indicate that the color is more yellow and green than blue and red. The distribution of color in the YCbCr space is illustrated in Figure 2.

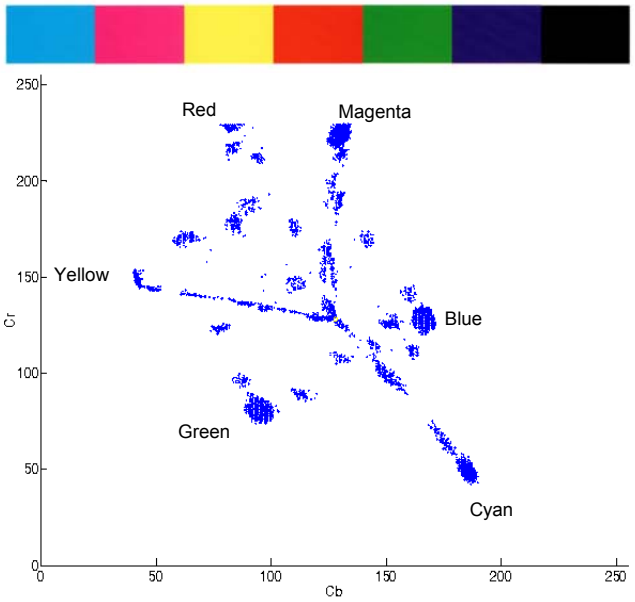

***Figure 2**. Top: RGB scan of solid patches of Cyan, Magenta, Yellow, Red, Green, Blue, and Black. Bottom: Cb and Cr components of the scan pixels.*

The neutral axis passes through the center of the plane, i.e., where *Cb* = 128 and *Cr* = 128. The most chromatic pixels are labeled and are the farthest from the center. Equivalent to metric chroma in the CIE L*a*b* color space, we define colorfulness of a pixel *P* as the Euclidean distance (ℓ_2 norm) from the neutral center in the CbCr plane as follows:

$$C(p) = \sqrt{(Cb(p) - 128)^2 + (Cr(p) - 128)^2}\,, \qquad (2)$$

where *Cb(p)* is the *Cb* component of pixel *P*, and *Cr(p)* is the *Cr* component. A less complex alternative is the Manhattan Distance (ℓ_1 norm) that can be written as:

$$C(p) = |Cb(p) - 128| + |Cr(p) - 128|\,. \qquad (3)$$

With these definitions of colorfulness, it is possible to classify a pixel as colorful or not given an appropriate threshold that is based on example scans of monochrome and color documents. The results in this paper are based on using Eq.(2).

Colorfulness Threshold Model

In order to classify a pixel as color or monochrome, we compute its colorfulness according to Eq.(2) and compare it to a threshold. A pixel is classified as color it if is more colorful than the threshold, and conversely, it is classified as monochrome if it is less colorful than the threshold. The choice of colorfulness threshold requires the consideration of variables such as, scanner

technology, media type, and expected document content. Rather than use trial and error in tuning the threshold, we modeled the cumulative colorfulness histogram of 16 color and monochrome test pages that are representative of the type of documents expected to be processed on the device. The document contents included images, graphics, line art, text, solid patches, and color ramps. Figure 3 shows the resulting cumulative colorfulness histogram. It is bimodal with a high peak indicating the presence of many pixels that are minimally colorful. The smaller peak shows the presence of colorful pixels that are less in number and with a wider colorfulness distribution.

The optimal threshold should separate the peaks, i.e. classes, while minimizing the probability that a pixel is misclassified given its colorfulness value. This is achieved through modeling the cumulative colorfulness histogram using a mixture of Gaussian distributions *P(x)*, as follows:

$$P(x) = \sum_{i=1}^{c} \pi_i \, p(x|C_i) \,, \tag{4}$$

where $0 < x < 128$ is the pixel colorfulness bin center, c is the number of normal mixtures (2 in this case), π_i is the mixing proportion of the C_i mixture, and the distributions are:

$$p(x|C_i) = \frac{1}{\sigma_i\sqrt{2\pi}} e^{-(x-\mu_i)^2/2\sigma_i^2} \,, \tag{5}$$

where $\theta_i=\{\mu_i,\sigma_i\}$ are C_i parameters. In order to estimate the mixture parameters (means, variances, and mixing proportions), we use the Expectation-Maximization (EM) algorithm [5, 6, 7]. The iterative algorithm can be described as having four main steps. First, the parameters must be initialized with a guess based on the histogram shape.

The second step is called the "E-Step". In it, δ_{ix}^k is computed at iteration k for every histogram bin x as follows:

$$\delta_{ix}^k = \frac{H[x]\,\pi_i^k\, p(x|\theta_i^k,C_i)}{\sum_{l=1}^{c} \pi_l^k\, p(x|\theta_l^k,C_l)} \,, \; 0 \le x < W \,, \tag{6}$$

where k is the iteration number, W is the number of bins, and $H[x]$ is the cumulative colorfulness histogram data to be modeled. Note that δ_{ix}^k represents the likelihood that data from a given bin is extracted from a specific distribution.

The third step is called the "M-Step". Here, the algorithm computes the new mean, variance, and new proportion of each mixture as follows:

$$\pi_i^{k+1} = \frac{\sum_{x=0}^{W-1} \delta_{ix}^k}{\sum_{x=0}^{W-1} H[x]} \,, \tag{7}$$

$$\mu_i^{k+1} = \frac{\sum_{x=0}^{W-1} \delta_{ix}^k \, x}{\sum_{l=1}^{c} \sum_{x=0}^{W-1} \delta_{lx}^k} \,, \tag{8}$$

and

$$(\sigma_i^{k+1})^2 = \frac{\sum_{x=0}^{W-1} \delta_{ix}^k \, (x-\mu_i^k)^2}{\sum_{l=1}^{c} \sum_{x=0}^{W-1} \delta_{lx}^k} \,. \tag{9}$$

Finally, the E-step and the M-step are repeated until the Gaussian means and variances converge, i.e., do not change significantly from one iteration to the next.

The results of applying EM with a mixture of two Gaussians to the cumulative colorfulness histogram data are shown in Figure 3. The cumulative colorfulness histogram data is shown in black. The peak between 0 and 10 is the contribution of all the less colorful, or monochrome, pixels in the test scans. The distribution of more colorful pixels lies between 10 and 50.

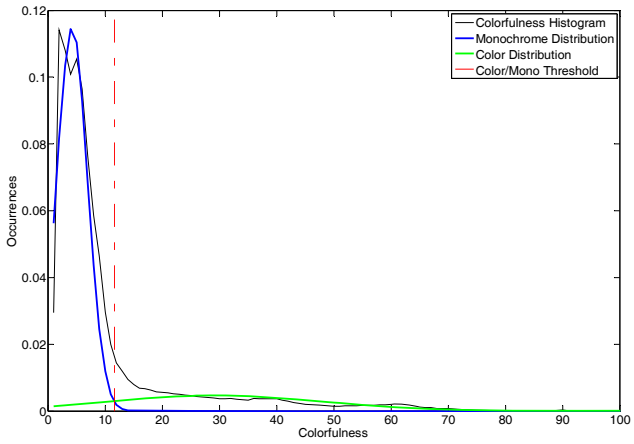

***Figure 3**. Gaussian Mixture model of the cumulative colorfulness histogram.*

A mixture of two Gaussians was chosen to model the histogram data since there are two classes of colorfulness of a pixel – colored and monochrome. Applying the EM algorithm with an initial guess of the distribution parameters yields the two Gaussians shown. The class of monochrome pixels (blue distribution) is centered at $\mu_1 = 4.24$, with a standard deviation of $\sigma_1 = 2.71$ and a $\pi_1 = 0.78$ mixing proportion. The class of color pixels (green distribution) has a mean of $\mu_2 = 29.35$, a standard deviation of $\sigma_2 = 18.85$, and a $\pi_2 = 0.22$ mixing proportion.

Given the Gaussian mixture model, it is possible to calculate the probability that a pixel is monochromatic or color given its colorfulness following Eq.(2). For example, a pixel with colorfulness equal to 20 is more likely to be a color pixel than a monochromatic one – according to the Gaussian mixture model.

For implementation in the scanner hardware, it is faster to simply use a threshold value for classification, such that a pixel is classified as color if its colorfulness is lager or equal to the threshold value and mono otherwise. The histogram model yields this threshold as being equal to *11.68*. This is the value where the two distributions intersect, as indicated by the vertical dashed line in Figure 3.

Algorithm Description

The proposed algorithm for detecting a monochrome page within a color copy job collects pixel colorfulness information from scan bands (that are divided into blocks) and analyzes the accumulated band data after the last scan band is processed. If enough local colorfulness is detected, the algorithm classifies the

scanned page as color; otherwise, it is designated as monochrome. A flow diagram showing the processing stages is given in Figure 4.

The scanner starts with the scan bar at the top of the page and collects scan band data as it moves down the page. In this experiment, the scan band pixel data is 8-bit RGB, where the scan band was 64 pixels high by 2560 pixels wide. The pixels are then converted to the 8-bit YCbCr color space using Eq.(1), and their colorfulness is computed according to Eq.(2). Using the colorfulness threshold, found via probability modeling in the "Color Processing" Section, the pixels are classified as either color or monochrome.

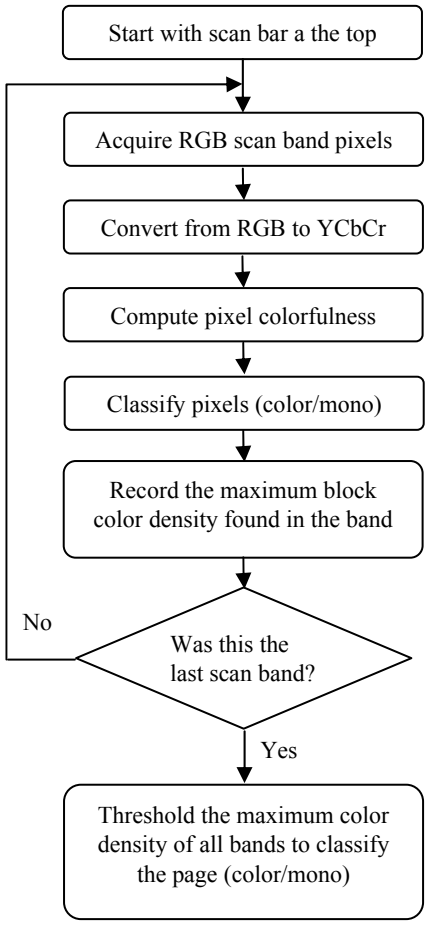

***Figure 4**. A Flow diagram of the monochrome page detection algorithm.*

The final scan band operation is to assess the amount of color density in the band. This step is designed to catch cases where a small amount of localized color is present in the band - a colored letter, for example. In order to detect local color, the band is divided into 64 by 64 blocks and the percentage of color pixels within them is computed. For the scan band, it is enough to save the highest color concentration (or density) amongst all its blocks. This ensures that a scan band will flag the presence of color even if it is localized in only one of its blocks. After the scan band is processed, the algorithm analyzes the amount of maximum color density across all bands to classify the page as either a color or a monochrome page. If color is present in a band, this metric will peak. If the maximum color density across all bands is below a threshold (we used 5%), the page is classified as monochrome. On the other hand, if the maximum band color density has peaks rising above the threshold, the page is classified as color.

Results

Misclassifying a color page as monochrome is worse than misclassifying a monochrome page as color, because even small amounts of color printed as monochrome are obvious. On the other hand, it is less obvious when monochrome content is copied as color [3]. For this reason, the classifier must be tuned to be sensitive to detect small amounts of color content, while still being robust to isolated color pixels from color noise. The example in Figure 5 illustrates some of these aspects. A raw CIS scan of an example page is shown in Figure 5 (a), in addition to tick marks along the left side indicating the scan band boundaries. The page is mostly monochromatic, but contains some color at the top right, in the center, and at the bottom. These colored pixels are shown in black in Figure 5 (b). Note the scattered color pixels in the center of the page – these are due to show-trough from the back of the page. Ideally, the algorithm should detect the presence of the small and large regions of color, but be robust to the scattered color content contributed by show-through and color noise.

Figure 5 (c) show a scatter plot of the *Cb* and *Cr* values of the test page pixels as a whole. The shown circle is centered at the neutral *(128,128)* point. The radius indicates the colorfulness threshold (of 11.68) that was determined through the Gaussian Mixture model. The pixels within the circle are deemed monochromatic, while the ones outside are classified are color pixels. There are a few color pixels that lie to the top-left (red) of the circle – these are from show-trough. Similarly, there are color pixels that lie to the bottom-right (cyan) of the threshold circle. These are more colorful, since they are further away from the neutral point, and more numerous – they are the color content.

As the scan bar moves down the page, the algorithm classifies all the scan band pixels using the colorfulness threshold. Every scan band (64 by 2560 pixels) is divided into blocks (64 by 64 pixels). The percentage of color pixels, i.e. color density, is computed for each block. If a scan band contains only scattered color pixels, then all the blocks will have a low color density. Conversely, a block will have a high color density if it contains a color object such as the letter "T" in this example page.

For every scan band, only the maximum color density is recorded for analysis after the last scan band is processed. This ensures a peaking signal for a band if it contains at least one color object. Figure 5 (d) shows the maximum color density over all the test page's bands. It also shows a threshold set to 5%, which was chosen by trial and error. This threshold allows the classification of the page as a whole as either a monochrome or color page. It can be seen, that the three locations of color content mentioned earlier (see Figure 5 (b)) produce peaks in the plot around the bands numbers 2, 32, and 50. Around band 24, there is a small peak that is only slightly larger than the threshold. The threshold can be easily tuned to a set of test documents, or can even be implemented on the MFP as a user setting to control the sensitivity of the monochrome page detection algorithm.

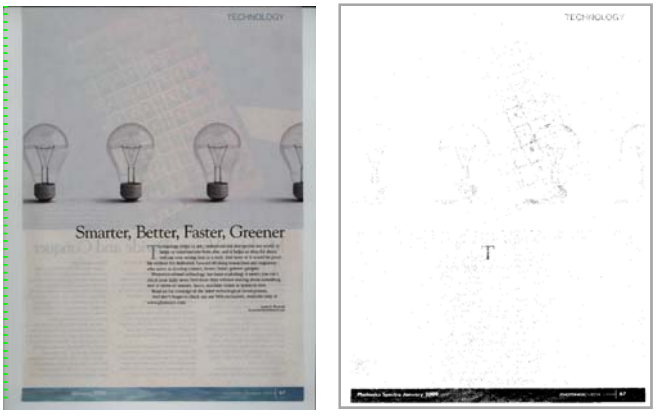

Figure 5. *(a) Raw CIS scan of a test page with scan band boundary marks (b) Pixels classified as colorful are set to black (c) Scatter plot of pixels on the CbCr plane (d) maximum block color density over scan bands.*

Conclusions

The presented algorithm serves to detect a monochrome (or neutral) page in a color copy job on CIS scanners that are part of multi function printers. The processing starts after the first scan band data is obtained and the maximum concentration of colorful pixels in each band is recorded to classify the page as color or monochrome after the scan is completed. Pixel colorfulness is defined in the YCbCr color space, since it presents an efficient and popular color encoding method. The modeling of the colorfulness of pixels contained within a number of test documents, employing the Expectation Maximization algorithm on a Gaussian Mixture model, yielded an optimal colorfulness threshold for pixel classification. The results indicate that the algorithm is robust to scanner color noise, while still being capable of detecting small concentrations of color in the scanned page.

References

[1] M. H. H. Brassé, S. P. R. C. de Smet, Data Path Design and Image Quality Aspects of the Next Generation Multifunctional Printer, Image Quality and System Performance V, Proc. SPIE, Vol. 6808, 68080V, (2008).

[2] H Nishida, T. Suzuki, A Multiscale Approach to Restoring Scanned Color Document Images with Show-through Effects, Seventh International Conference on Document Analysis and Recognition, Vol.1 Issue, 3-6 Aug. pp. 584 – 588, (2003).

[3] X. Dong, K. Hua, P. Majewicz, G. McNutt, C. A. Bouman, J. P. Allebach, I. Pollak, "Document Page Classification Algorithms in Low-end Copy Pipeline," Journal of Electronic Imaging, 17(4), 043011, (2008).

[4] Y. Hirota, K. Toyama, S. Imaizumi, H. Hashimoto, K. Ishiguro, Image Processing Apparatus, Image Forming Apparatus and Color Image Determination Method Thereof, U.S. Patent No. 7319786, (2008).

[5] J. C. Handley, Scanned Color Document Image Segmentation Using the EM Algorithm, International Congress of Imaging Sciences, pp. 675-678, (2006).

[6] Geoffrey J. McLachlan and Thriyambakam Krishnan, The EM Algorithm and Extensions, 2nd Edition, (Wiley, NY 2008).

[7] A. P. Dempster, N. M. Laird, D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm". Journal of the Royal Statistical Society. Series B, Vol. 39, No. 1, pp. 1–38, (1977).

[8] Recommendation ITU-R BT.601-6, "Studio Encoding Parameters of Digital Television for Standard 4:3 and Wide Screen 16:9 Aspect Ratios," International Telecommunications Union, (1982, 1986, 1990, 1992, 1994, 1995, 2007).

Author Biography

Nathir A. Rawashdeh received the Ph.D. and B.S. degrees in electrical engineering from the University of Kentucky in 2007 and 2000 respectively. He received the M.S. in electrical and computer engineering from the University of Massachusetts, Amherst in 2003. Currently, he is a Senior Software Engineer in the Color Science and Imaging Department at Lexmark International. He was with The MathWorks between 2000 and 2003, where he worked on DSP-target tools. His research interests include image quality analysis, color science, and pattern recognition.
Dr. Rawashdeh is a member of the IEEE, ARRL and SPIE.