Text Line Detection Based on Cost Optimized Local Text Line Direction Estimation *

Yandong Guo[a]†, Yufang Sun[a], Peter Bauer[b], Jan P. Allebach[a] and Charles A. Bouman[a]

[a]School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, 47907;
[b]Hewlett-Packard Co., P, Boise, ID, 83714

ABSTRACT

Text line detection is a critical step for applications in document image processing. In this paper, we propose a novel text line detection method. First, the connected components are extracted from the image as symbols. Then, we estimate the direction of the text line in multiple local regions. This estimation is, for the first time, to our knowledge, formulated in a cost optimization framework. We also propose an efficient way to solve this optimization problem. Afterwards, we consider symbols as nodes in a graph, and connect symbols based on the local text line direction estimation results. Last, we detect the text lines by separating the graph into subgraphs according to the nodes' connectivities. Preliminary experimental results demonstrate that our proposed method is very robust to non-uniform skew within text lines, variability of font sizes, and complex structures of layout. Our new method works well for documents captured with flat-bed and sheet-fed scanners, mobile phone cameras, and with other general imaging assets.

Keywords: Text line detection, cost optimization, graphical model, message passing, image segmentation

1. INTRODUCTION

Text line detection is a critical step for tasks such as document layout analysis.[1,2] Text line detection with low computational cost and high accuracy is still considered as an open problem for complex document images or natural images. The complexity of document images comes from irregular layout structure, a mixture of machine printed and hand written characters, and/or the variability of text line directions.

Many methods have been proposed for accurate text line detection. One category of popular methods are top-down approaches, such as the Hough transform-based methods.[3–5] Hough transform-based methods detect text lines using the "hypothesis-validation" strategy: The potential text lines (collinear alignments of symbols), are hypothesized in the Hough domain, and validated in the image domain. This "hypothesis-validation" strategy is computationally expensive. Moreover, in order to deal with non-straight text lines, or complex layout structure, extra pre-processing and/or post-processing strategies are needed to make this method robust.

Another category of popular methods are the smearing methods in,[6,7] which follow a bottom-up scheme. The basic idea is to grow the text line region by recursively finding and incorporating the closest characters. Compared with Hough transform-based methods, smearing methods can deal with fluctuating lines better. However, only searching the closest characters limits the information to a very small region, and is sensitive to noise. Typically, the smearing methods contain parameters that need to be accurately and dynamically tuned. A nice review of the text line detection methods can be found in.[8–11]

We also notice that a lot of pre/post processing methods have recently been proposed to improve the text line detection performance. One good example about pre-processing is the edge-enhancement-based connect component extraction.[1,12] In the same paper,[1] a text line is rejected (as post processing) if a significant portion of the objects in this text line are

*This work was supported by the Hewlett-Packard Company.

†Yandong Guo now is an Associate Researcher at Microsoft, Redmond, WA 98052; Email address: yag@microsoft.com
Further author information: (Send correspondence to Yandong Guo)
Yandong Guo: E-mail: yag@microsoft.com, Telephone: +1 425-538-6224
Yufang Sun: E-mail: sun361@purdue.edu, Telephone: +1 765-494-3518
Peter Bauer: E-mail: peter.bauer@hp.com, Telephone: +1 208 396 6981
Jan P. Allebach: E-mail: allebach@purdue.edu, Telephone: +1 765 494 3535
Charles A. Bouman: E-mail: bouman@purdue.edu, Telephone: +1 765 494 0340

Color Imaging XX: Displaying, Processing, Hardcopy, and Applications, edited by Reiner Eschbach, Gabriel G. Marcu, Alessandro Rizzi, Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 9395, 939507 · © 2015 SPIE-IS&T
CCC code: 0277-786X/15/$18 · doi: 10.1117/12.2083709

repetitive. Our paper is purely focused on the text line detection, but can also potentially incorporate these pre/post processing steps.

In this paper, we propose a novel text line detection method. In the first step, we design a cost function to estimate the local text line direction and the collinearity relationship for character pairs within the local region. We also propose an efficient way to optimize this cost function. Our cost function is designed based on the observation that a text line is typically formed by a set of characters densely distributed along a smooth curve. Since the curve of a text line is typically smooth, within a relatively small region, we can approximately model the text line as a straight line segment. Moreover, since there are more characters along the direction of the text line than there are in other directions, we design our cost function in a way that minimizing the cost encourages the local text line to contain as many characters as possible. In the second step, we propose a method called graphical model-based text line construction (GMTC). In GMTC, we build a graphical model by considering each of the character as a node, and then, we group characters into text lines by separating the graph into subgraphs based on the estimation results in the first step (local text line direction and the collinearity of characters within local regions).

Our method is different from the previous methods from the following perspectives. Unlike the typical Hough transform-based methods,[3–5] we model the text line as the concatenation of multiple small line segments to approximate a smooth curve (rather than as a global straight line) to better handle fluctuating lines. Another difference is that our method efficiently optimizes a cost function to estimate the collinear alignments of characters rather than using a "hypothesis-validation" strategy. The difference from the typical smearing methods[6,7] is that, when we cluster characters into text lines, we impose constraints onto the variations of the directions of the local text lines to be merged (rather than being purely based on local character proximity), which makes our method less sensitive to noise. Moreover, our model is different from the graphical models previously used for text line detection since the messages passed between nodes in our model are from the local collinearity obtained by using our cost optimized local text line direction estimation (C-LTDE) method. Experiments with a variety of images demonstrate that the proposed method is very fast and robust to non-uniform skew within text lines, variability of font sizes, and complex structures of layout.

In Sec. 2, we present our text line detection method. In Sec. 3, we present experimental results.

2. TEXT LINE DETECTION

In this section, we describe our text line detection method. First, we extract the N connected components, called symbols (characters), from the image, denoted by $\{\mathbf{s}_i\}_{i=1}^{N}$.[13–15] The centroid of the i^{th} symbol is denoted by $\mathbf{x}_i = (x_{i,1}, x_{i,2}, 1)^T$, where $x_{i,1}$ and $x_{i,2}$ are its horizontal and vertical coordinates, respectively. In this paper, we use homogeneous coordinates to introduce compactness in our formulation. Then, the text line is constructed according to the geometric locations of the centroids of the symbols, with details described in the following subsections.

2.1 Local text line direction estimation

Although text lines may contain non-uniform skew, within a relatively small region, the centroids of the symbols in the same text line tend to fall on a straight line. Therefore, we estimate the directions of the text lines in different local small regions separately.

More precisely, we go through all the symbols. For the i^{th} symbol, we define a local region centered at its centroid $\mathbf{x}_i$. All the symbols in this local region are denoted by $\mathbf{s}_{\partial i}$. An example is shown in Fig. 1. In the local region centered at $\mathbf{x}_i$, the direction of the text line containing $\mathbf{s}_i$ is denoted as θ_i, while the location of the text line is controlled by a scalar $\beta_{i,3}$. Following the homogenous coordinates used in the definition of $\mathbf{x}_i$, we define a vector

$$\boldsymbol{\beta}_i = [\beta_{i,1}, \beta_{i,2}, \beta_{i,3}]^T \,, \tag{1}$$

with the constraints,

$$\beta_{i,1} = \cos\theta_i \,, \tag{2}$$

$$\beta_{i,2} = \sin\theta_i \,. \tag{3}$$

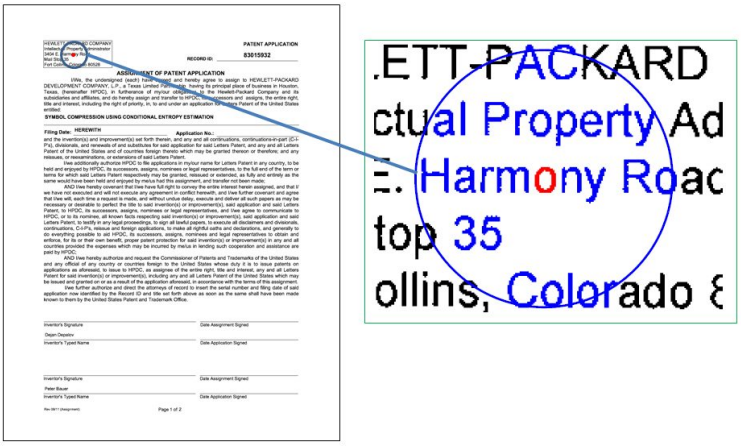


Figure 1. We estimate the direction of text lines in multiple local regions. The left figure is the original document image, while the right figure is obtained by zooming in this document image. This figure shows an example of the local region we use. For each given symbol $\mathbf{s}_i$, e.g. the letter "o" colored red in the figure, we define a small region centered at its centroid. The local direction of the text line containing $\mathbf{s}_i$ is estimated according to the location of the symbols $\mathbf{s}_i$ and $\mathbf{s}_{\partial i}$ in the local region. Here, the term ∂i denotes the indexes of the symbols within the local region centered at the centroid of $\mathbf{s}_i$. As shown in the figure, the set of symbols $\mathbf{s}_{\partial i}$ are colored blue.

With this homogenous framework, it can be shown that the distance between the centroid $\mathbf{x}_j$ and the straight line represented by $\boldsymbol{\beta}_i$ is

$$d(\mathbf{x}_j, \boldsymbol{\beta}_i) = |\boldsymbol{\beta}_i^T \mathbf{x}_j| \,. \tag{4}$$

In addition to the above assumption on the local collinearity of symbol centroids, we estimate $\boldsymbol{\beta}_i$ based on the assumption that the symbol density is higher along the text line direction than it is in other directions. Since not all the symbols in $\mathbf{s}_{\partial i}$ belong to the same text line, we propose a binary variable $\lambda_{i,j}$ to describe whether the j^{th} symbol belongs to the text line containing the i^{th} symbol.

$$\lambda_{i,j} = 1 \leftrightarrow \mathbf{s}_j \text{ belongs to the text line containing } \mathbf{s}_i;$$
$$\lambda_{i,j} = 0 \leftrightarrow \mathbf{s}_j \text{ does not belong to the text line containing } \mathbf{s}_i.$$

With the notation above, we design the following cost function to estimate $\boldsymbol{\beta}_i$ and $\boldsymbol{\lambda}_i = \{\lambda_{i,j} | j \in \partial i\}$,

$$\left\{\hat{\boldsymbol{\beta}}_i, \hat{\boldsymbol{\lambda}}_i\right\} = \arg\min_{\boldsymbol{\beta}_i, \boldsymbol{\lambda}_i} \left\{ \sum_{j \in \partial i} \alpha_j \left|1 - \lambda_{i,j}\right| + d(\mathbf{x}_i, \boldsymbol{\beta}_i)^2 + \sum_{j \in \partial i} \lambda_{i,j} d(\mathbf{x}_j, \boldsymbol{\beta}_i)^2 \right\}, \tag{5}$$

where the term $d(\mathbf{x}_i, \boldsymbol{\beta}_i)^2 + \sum_{j \in \partial i} \lambda_{i,j} d(\mathbf{x}_j, \boldsymbol{\beta}_i)^2$ accounts for the local collinearity assumption. During the minimization procedure, this term produces the estimation of $\boldsymbol{\beta}_i$ by fitting a straight line onto the centroids of symbols with non-zero $\lambda_{i,j}$. Simultaneously, this term tends to prune out symbols from the local text line by encouraging $\lambda_{i,j}$ to be 0.

On the other hand, the penalty term $\sum_{j \in \partial i} \alpha_j \left|1 - \lambda_{i,j}\right|$, encourages the local text line to contain as many symbols as possible. Without this term, after the minimization, none of the symbols in $\mathbf{s}_{\partial i}$ will be considered to be in the text line containing $\mathbf{s}_i$, since purely minimizing the second line of (5) would make all the $\lambda_{i,j}$ to be 0. The strength of the penalty term is controlled by the parameter α_j. We design the parameter α_j to introduce geometric meaning to our cost function. Suppose the width and height of the j^{th} symbol are denoted by w_j and h_j, respectively. The parameter α_j is calculated as

$$\alpha_j = \left(\frac{\max\{w_j, h_j\}}{2}\right)^2. \tag{6}$$

The way we calculate α_j has the following geometric interpretation. Calculating the partial derivative of (5) according to

$\lambda_{i,j}$, we get

$$\text{If } d(\mathbf{x}_j, \boldsymbol{\beta}_i) \leq \frac{\max\{w_j, h_j\}}{2} \text{, then } \lambda_{i,j} = 1 \tag{7}$$

$$\text{If } d(\mathbf{x}_j, \boldsymbol{\beta}_i) > \frac{\max\{w_j, h_j\}}{2} \text{, then } \lambda_{i,j} = 0\,. \tag{8}$$

The result in (7) shows that if the straight line $\boldsymbol{\beta}_i$ passes through the symbol $\mathbf{s}_j$, $\lambda_{i,j} = 1$, which means that the location of the symbol $\mathbf{s}_j$ contributes to the estimation of $\boldsymbol{\beta}_i$. On the contrary, as shown in (8), if the straight line $\boldsymbol{\beta}_i$ does not cut through the symbol $\mathbf{s}_j$, $\lambda_{i,j} = 0$, which means the symbol $\mathbf{s}_j$ is not considered to belong to the text line with $\mathbf{s}_i$; and the location of $\mathbf{s}_j$ will not contribute to the estimation of $\boldsymbol{\beta}_i$.

We design an alternating optimization scheme to solve (5). Multiple initial conditions are applied to handle the non-convexity of the cost function; and the matrix transform strategy in[16,17] is applied to improve the computational efficiency. For simplicity, we do not describe the details of the optimization here. After we obtain $\hat{\boldsymbol{\beta}}_i$, we calculate the direction of the local text line containing the symbol $\mathbf{s}_i$ as

$$\hat{\theta}_i = \arctan(\frac{\hat{\beta}_{i,2}}{\hat{\beta}_{i,1}})\,. \tag{9}$$

2.2 Text line construction

After we obtain $\hat{\boldsymbol{\beta}}_i$ and $\hat{\boldsymbol{\lambda}}_i$, $i = 1, \ldots, N$, we set up a graphical model $G = (V, E)$ to cluster the symbols into text lines. Here, the terms V and E denote the set of nodes and the set of edges, respectively. Each of the symbols is considered as a node in the graph. For any two symbol nodes $\mathbf{s}_i$ and $\mathbf{s}_j$, the probability that the symbol nodes $\mathbf{s}_i$ and $\mathbf{s}_j$ belong to the same text line is calculated as

$$p_{i,j} = \lambda_{i,j}\lambda_{j,i}\delta(\theta_i, \theta_j)\,. \tag{10}$$

In order to obtain a very efficient method, we here design $\delta(\theta_i, \theta_j)$ to have a binary output.

$$\begin{aligned} \delta(\theta_i, \theta_j) &= 1, \text{if } |\theta_i - \theta_j| \leq \theta_{\max}\,, \\ &= 0, \text{otherwise}\,. \end{aligned} \tag{11}$$

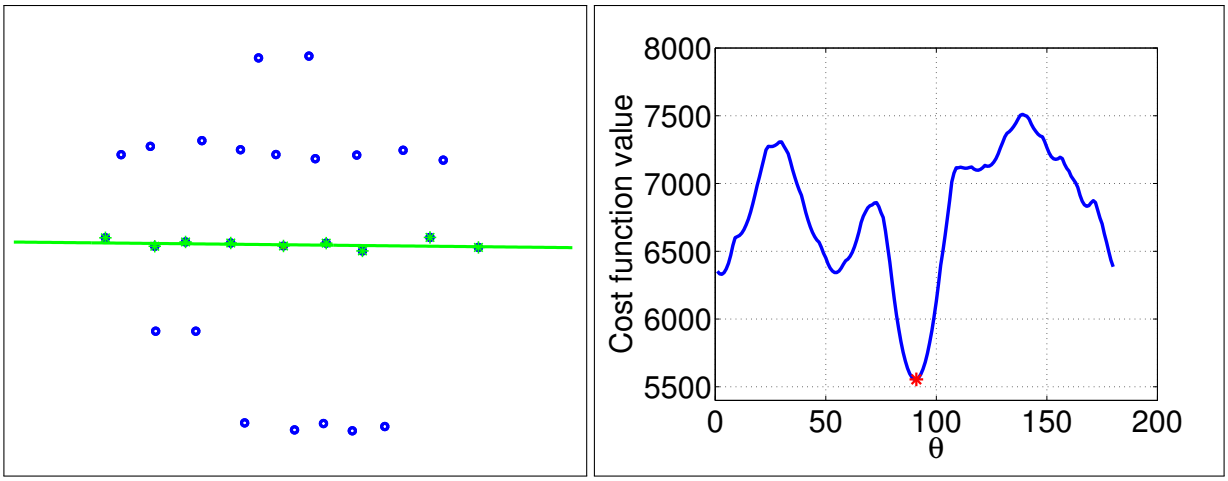
Empirically, we set $\theta_{\max} = \frac{\pi}{6}$. At the cost of greater computation, a continuous output model for $\delta(\theta_i, \theta_j)$ with a more sophisticated graph inference methods, such as loopy belief propagation or expectation propagation[18–20] could improve the robustness of our method.

With the graph G constructed, we construct text lines using the following procedure:

1. Mark all the symbol nodes as unclustered.
2. Pick any unclustered symbol node in the graph G as the source node, and find all the reachable symbol nodes using the breadth-first search (BFS) algorithm.[21] These reachable symbols and the source symbol are considered to belong to the same text line.
3. Mark these symbols as clustered.
4. Repeat Step 2 and Step 3 until all the symbols are clustered.

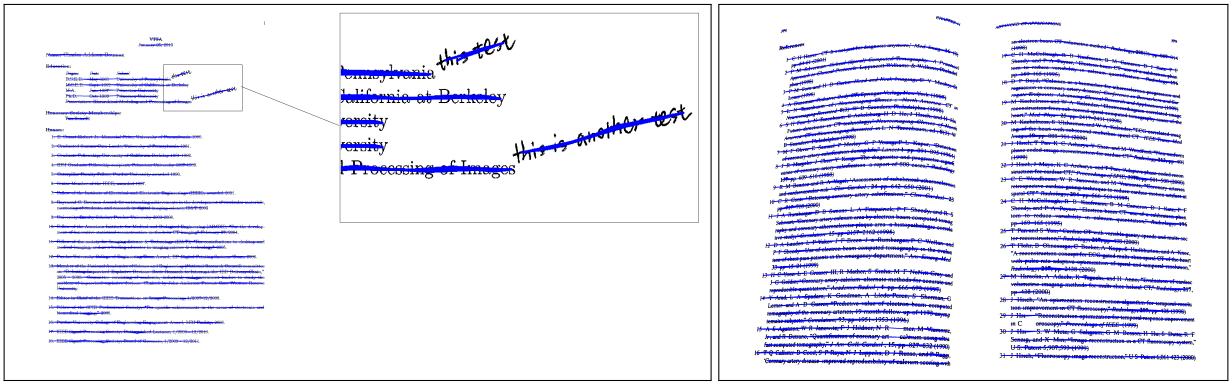
3. EXPERIMENTAL RESULTS

We conducted experiments with a variety of document images to demonstrate the effectiveness of our text line detection method. We also demonstrate that the proposed method can be applied to improve the accuracy of a state-of-art text detection method.[13]



(a) Symbol centroids in local region (b) Cost value in (5) with different $\boldsymbol{\beta}$ values

Figure 2. Example of the local text line direction estimation. In subfigure (a), each of the dots represents the centroid of a symbol in the local region shown in Fig. 1 (b). Fixing $\boldsymbol{\beta}_i$ with different values, we minimized the cost function in (5) over $\boldsymbol{\lambda}_i$, and show the value of the minimized cost function in subfigure (b). As indicated in the figure, the best value for θ_i is 90.6°, corresponding to the green line in subfigure (a).



(a) Test case 1 (b) Test case 2

Figure 3. Text line detection results obtained with our method. The symbols in the same text line are connected using a blue line segment.

3.1 Local text line direction estimation

In this subsection, we show an example to demonstrate our local text line direction estimation. The local region used in this subsection is shown in Fig. 1 (b); and the centroids of the symbols in this region are shown in Fig. 2 (a).

In our experiment, we obtained the value of the cost function in (5) with different $\boldsymbol{\beta}_i$ using the following scheme. We gradually changed the value of θ_i from 1° to 180°. For each fixed $\boldsymbol{\beta}_i = [\cos\theta_i, \sin\theta_i, \beta_{i,3}]^T$, we minimized (5) over $\boldsymbol{\lambda}_i$. The minimum values of the cost function given different θ_i values are shown in Fig. 2(b). As indicated in the figure, when $\theta = 90.6^\circ$, the overall minimum value was obtained, which corresponds to the straight line shown as the green line in Fig. 2 (a).

3.2 Text line detection in complex document images

In this subsection, we conduct experiments with complex document images to demonstrate the robustness of our algorithm. For brevity, we only discuss some of the challenging cases in Fig. 3. Figure 3 (a) shows a difficult case for smearing methods. Since the smearing methods keep trying to find the closest symbols to merge, the closely located hand written text line and the printed text line tend to be considered as the same text line. But as shown in Fig. 3 (a), our algorithm successfully separated the hand written text line from the machine printed text lines. This is because our method connects

symbols incorporating to the local text line direction information. Another example is shown in Fig. 3 (b), which shows that our method can detect text lines that are not straight in a complex layout structure, while a typical Hough-based method cannot.

4. CONCLUSION

In this paper, we proposed a text line detection method based on local text line direction estimation. Our method has the following advantages over existing methods: First, we estimate the direction of the text line within small local regions, which makes our method more robust and able to solve the non-uniform skew issue. Then, we connect symbols into text lines based on the connectivity and the local text line direction information. This is an advantage over purely bottom-up approaches, such as the smearing method, because for two given symbols, we consider them to be in the same text line not only based on the distance between them, but also the local information of both of the two symbols (local text line direction). Experimental results demonstrate the effectiveness of our method.

REFERENCES

[1] Chen, H., Tsai, S. S., Schroth, G., Chen, D. M., Grzeszczuk, R., and Girod, B., "Robust text detection in natural images with edge-enhanced maximally stable extremal regions," in [*Proc. of IEEE Int'l Conf. on Image Proc.*], (2011).

[2] Lu, C., Wagner, J., Pitta, B., Larson, D., and Allebach, J., "SVM-based automatic scanned image classification with quick decision capability," in [*Proc. SPIE 9015, Color Imaging XIX: Displaying, Processing, Hardcopy, and Applications*], (2014).

[3] Fletcher, L. and Kasturi, R., "A robust algorithm for text string separation from mixed text/graphics images," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **10**, 910–918 (Nov. 1988).

[4] Likforman-Sulem, L., Hanimyan, A., and Faure, C., "A Hough based algorithm for extracting text lines in handwritten documents," in [*ICDAR*], 774–777 (1995).

[5] Pu, Y. and Shi, Z., "A natural learning algorithm based on Hough transform for text lines extraction in handwritten documents," in [*Proc. of the 6 Int'l Workshop on Frontiers in Handwriting Recognition*], 637–646 (1998).

[6] Shi, Z. and Govindaraju, V., "Line separation for complex document images using fuzzy runlength," in [*DIAL*], 306–313 (2004).

[7] Bourgeois, F. L., Emptoz, H., Trinh, E., and Duong, J., "Networking digital document images," in [*ICDAR*], 379–383 (2001).

[8] Louloudisa, G., Gatosb, B., Pratikakisb, I., and Halatsisa, C., "Text line detection in handwritten documents," *Pattern Recognition* **41**, 3758–3772 (2008).

[9] Louloudis, G., Gatos, B., and Halatsis, C., "Text line detection in unconstrained handwritten documents using a block-based hough transform approach," in [*ICDAR*], **2**, 599–603 (2007).

[10] Li, Y., Zheng, Y., and Doermann, D., "Detecting text lines in handwritten documents," in [*Proc. of IEEE Int'l Conf. on Pattern Recognition*], **2**, 1030–1033 (2006).

[11] Yi, C. and Tian, Y., "Text string detection from natural scenes by structure-based partition and grouping," *IEEE Trans. on Image Processing* **20**(9), 2594–2605 (2011).

[12] Grzeszczuk, R., Chandrasekhar, V., Takacs, G., and Girod, B., "Method and apparatus for representing and identifying feature descriptors utilizing a compressed histogram of gradients," (May 20 2010). WO Patent App. PCT/IB2009/007,434.

[13] Haneda, E. and Bouman, C. A., "Text segmentation for MRC document compression," *IEEE Trans. on Image Processing* **20**, 1611–1626 (2011).

[14] Guo, Y., Depalov, D., Bauer, P., Bradburn, B., Allebach, J. P., and Bouman, C. A., "Binary image compression using conditional entropy-based dictionary design and indexing," in [*Proc. SPIE 8652, Color Imaging XIII: Displaying, Processing, Hardcopy, and Applications*], **8652**, 865208–1–865208–10 (2013).

[15] Guo, Y., Depalov, D., Bauer, P., Bradburn, B., Allebach, J. P., and Bouman, C. A., "Dynamic hierarchical dictionary design for multi-page binary document image compression," in [*Proc. of IEEE Int'l Conf. on Image Proc.*], (2013).

[16] Cao, G. and Bouman, C., "Covariance estimation for high dimensional data vectors using the sparse matrix transform," in [*Advances in Neural Information Processing Systems*], 225–232 (2008).

[17] Cao, G., Guo, Y., and Bouman, C. A., "High dimensional regression using the sparse matrix transform (SMT)," in [*Proc. of IEEE Int'l Conf. on Acoust., Speech and Sig. Proc.*], 1870–1873 (2010).

[18] Frey, B. and MacKay, D., "A revolution: Belief propagation in graphs with cycles," in [*Advances In Neural Information Processing Systems*], 479–485 (1998).

[19] Minka, T. P., *A family of algorithms for approximate Bayesian inference*, Ph.D. dissertation, Massachusetts Institute of Technology (2001).

[20] Qi, Y. and Guo, Y., "Message passing with l_1 penalized KL minimization," in [*Proceedings of the* $30th$ *International Conference on Machine Learning, and Journal of MLR*], **28**, 262–270 (2013).

[21] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C., [*Introduction to Algorithms*], The MIT Press, 2 ed. (2001).