https://doi.org/10.2352/ISSN.2470-1173.2019.10.IQSP-300

Block window method with logistic regression algorithm for streak detection*

Runzhe Zhang [a], *Eric Maggard* [b], *Renee Jessome* [b], *Yousun Bang* [c], *Minki Cho* [c], *Jan Allebach* [a]
[a] School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47906, U.S.A.
[b] HP Inc., Boise, ID 83714, U.S.A.
[c] HP Inc., Suwon City, KOREA.

Abstract

Streaks are one of the most common defects in electrophotographic printers, and dramatically affect print quality. Researchers have developed methods to detect streaks. Then, using the detection result helps us to diagnose issues of the printer and discover broken components of the electrophotographic printer. In previous work, the streak detection methods are based on a particular printer or particular streak defects, such as Intermediate Transfer Belt (ITB) or Organic Photoconductor (OPC) drum streak. In this paper, we design a Block Window Method to pre-test the images with streak defects. It is based on the local ΔE value in a block window and works for different kinds of streaks. After using the Block Window Method, the detection result includes small streaks or noise defects that are too localized for humans to see. We use the logistic regression algorithm to classify the real visible streaks and small invisible streaks. This process will improve the accuracy of the detection result. After the classification, we can get the streak detection result, which is significant for extracting the feature vector of the streak defects in the test image. Then, we can use the feature vector to classify different streak defects.

1. Introduction

Page quality (PQ) is one of the most significant issues with electrophotographic printers. There are many reasons for PQ issues, such as limitations of the electrophotographic process, faulty printer components, or other failures of the print mechanism. These reasons can produce different kinds of PQ issues, like streaks, bands, and gray spots. In this paper, we propose a method for detecting streak defects. The major axis of streaks is along the printing process direction. They occur when the ITB (Intermediate Transfer Belt), OPC (Organic Photo Conductor), or other color cartridge components include issues. Through analyzing streaks detection result, we can diagnose the issues of the printer. Figure 1 is the example of a streak defect.

Our streak detection algorithm is based on the analysis of a standard printed test page, such as is shown in Figure 1. This page could be printed by the customer, or an on-site field technician, scanned at the customer's location, and then analyzed by the printer firmware. Based on the detection result, our algorithm can analyze the issues of the printer and send a report to printer customer service. The customer service will analyze the printer problems without using customer images.

This new method for streak defect detection is based on recent image quality work. These techniques focus on measurement of print quality, and address assessment of page non-uniformity [3]-[8], fine-pitch banding [9]-[11] and some image processing methods [12]-[14].

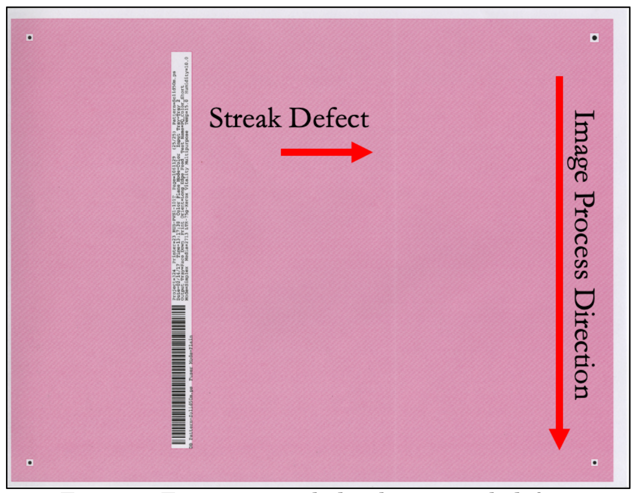


Figure 1: Test image with developer streak defect.

In this paper, we build a solution to find vertical streaks in test images. We will cover the proposed procedure, the results, and the conclusions in the following sections.

2. Streak detection procedure

In this section, we introduce the details of detecting the streak defects in the customer images. The overall pipeline of the proposed method is shown in Figure 2. It will be presented in two parts: 1. Using the Block Window Method to get the streaks detection result including noise; 2. Using Logistic Regression to remove noise detections.

2.1 Block window method

The block window method is to separate the test image into several blocks, and analyze the ΔE value of each block to detect the streaks.

2.1.1 Pre-processing

The scanned test images usually are saved in *sRGB* color space and are gamma corrected. Firstly, the test images should be gamma un-corrected and color space converted

*Research supported by HP Inc., Boise, ID 83714 from $sRGB$ to $CIE\ L^*a^*b^*$. Secondly, we use a Gaussian Filter (descreening algorithm) to remove halftone patterns. The Gaussian Filter size is 15×15 (pixels) with $\sigma = 2$. An original image and the corresponding descreened image are illustrated in Figure 3a and Figure 3b. The descreening equation is:

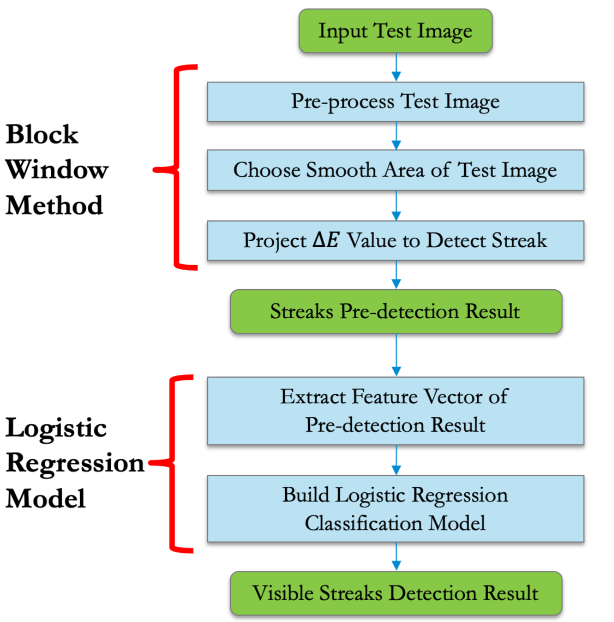
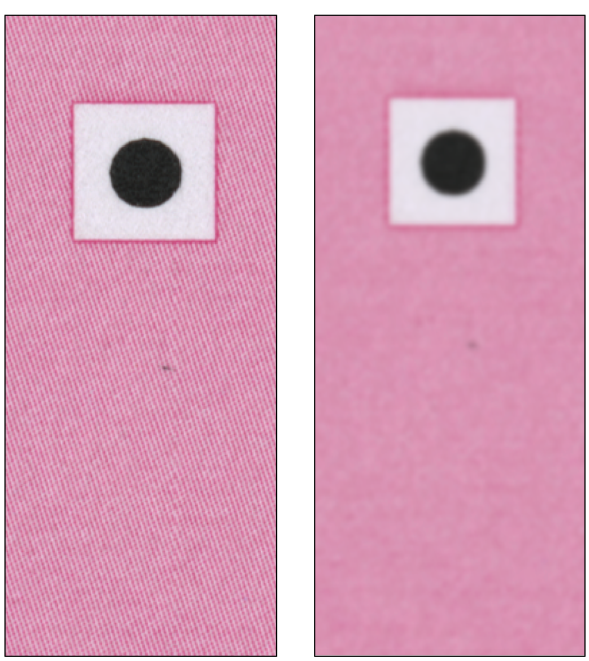


Figure 2: The overall pipeline of the streak detection method.



a. Original test image b. De-screened test image

Figure 3: The result of the descreening test image.

$$f'(x,y,c) = \sum_{i=-7}^{7}\sum_{j=-7}^{7} f(x+i,y+j,c)g(i,j) \tag{1}$$

The image $f(x,y,c)$ is the input in $CIE\ L^*a^*b^*$ color space, $g(i,j)$ is the 15×15 Gaussian Filter, and $f'(x,y,c)$ is the descreened result.

After getting the descreened $CIE\ L^*a^*b^*$ image, we should separate the image into many blocks. In this experiment, we set the block size to 200×200 (pixels). For different test images, we can use different block sizes to choose the smooth area.

2.1.2 Choose smooth area

After the preprocessing step, we get the block descreened test image in $CIE\ L^*a^*b^*$ color space. For the next step, we should choose the smooth area of the test image, as there is some custom content in the test image. This custom content may include vertical lines. To select the smooth area of the test image, we use the average ΔE value of each block. ΔE is a metric for understanding how the human eye perceives color differences.

Firstly, we calculate the ΔE of each pixel in one block. The subscript (i,j) means the pixel position in the image. The subscript ave means the average value of each channel value in one block. M is the block size.

Figure 4: The smooth area of the test image.

$$\Delta E_{(i,j)} = \sqrt{(L_{(i,j)} - L_{ave})^2 + (a_{(i,j)} - a_{ave})^2 + (b_{(i,j)} - b_{ave})^2} \tag{2}$$

$$L_{ave} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{M} L_{(i,j)}}{M^2}$$

$$a_{ave} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{M} a_{(i,j)}}{M^2}$$

$$b_{ave} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{M} b_{(i,j)}}{M^2}$$

Secondly, we use ΔE value of every pixel in one block to calculate the average ΔE value of the block.

$$\Delta E_{block} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{M}\Delta E_{(i,j)}}{M^2}$$

After getting all the ΔE_{block} values of the test image, we set a threshold for ΔE_{block}, and choose the smooth area of the test image. This threshold is an experimental parameter. If the threshold is too small, streak defects will be included in the non-smooth areas. Otherwise, many small customer contents will belong to the smooth area. In our experiments, we checked 50 test images and chose the threshold to be 2. It can be verified by the streak feature in the Section 2.2.1. Figure 4 shows the smooth area result, where the white area of the mask image is the smooth areas.

2.1.3 Detect streaks using ΔE projection

For the next step, we use a ΔE projection in each smooth block to detect streaks. As streaks are the vertical lines in the test images (parallel to the printing process direction), we project the ΔE of each column in the block along the printing process direction.

Typically, the magnitude of the ΔE streak projections is very high (almost twice projection value of the smooth areas). To distinguish the streak from the smooth area, we set a threshold $T_{\Delta Eproj}$ according to Eq. (3). Only the projection magnitudes that exceed the threshold will be marked as steak defects.

$$T_{\Delta Eproj} = \Delta E_{block} \times M \tag{3}$$

After calculating the whole test image, we can get the streak pre-detection result as illustrated in Figure 5. The pre-detection result includes all the streaks visible to the naked eye. But it also contains many noise detections, which are labeled as streaks, even though they are too small to be seen by the human eye. We can compare Figure 1 and Figure 5 to find there is a noise detection in Figure 5. So, we use the logistic regression algorithm to remove the noise detections.

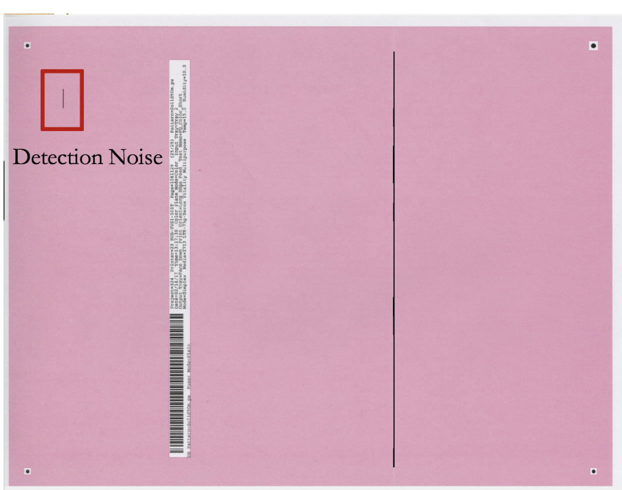


Figure 5: The block window method detection result.

2.2 Logistic regression application

In the Block Window Method, the threshold for determining smooth areas is an experimental parameter. To get more accurate streaks detection results for different kinds of streak defect test images, a better way is to use a different threshold parameter for each type of streak defect. In practice, we use one threshold for all kinds of streak defects. The detection result will include many noise detections for some types of streak defects. The way we remove the noise detections is by using the logistic regression method to classify the visible streak defects and the invisible streak defects, as illustrated in Figure 5.

2.2.1 Extract feature vector

For the logistic regression process, we developed five features to distinguish the visible streaks and invisible streaks. Before extracting the features, we use the connected component algorithm to label the different streak detection results. We introduce the five features below.

1. **Area of each streak detection result**
   This equals the number of pixels included in each streak detecion result, denoted by N.

2. **Length-width ratio of each streak detection result**
   Because the streak is a vertical line in the test image, the greater the length-width ratio of the detection result, the greater the probability of a streak defect. The length and width of each streak detection result are available from the connected component calculation process.

3. **Severity of each streak detection result**
   There are two factors that affect the severity of the streak detection: the area of the streak detection result and the average ΔE value of the streak detection. We calculate the severity of the streak detection according to Eq. (4).

$$S = \Delta E_{streak-ave} \times N \tag{4}$$

4. **L^* channel difference**
   We use lightness to classify streaks: light streaks and dark streaks. Because the L^* channel represents the lightness value and the most obvious characteristic to distinguish between streaks and background is lightness, we calculate the average value of the L^* channel in the smooth area and denote it as $\bar{L}_{smooth}$. Then, we use the Eq. (5) to calculate the L^* channel difference from the mean of each streak detection result.

$$\bar{L}_{streak} = \frac{\sum_{i=1}^{N}(L_i - \bar{L}_{smooth})}{N} \tag{5}$$

   If $\bar{L}_{streak}$ is positive, the streak is a light streak; otherwise, the streak is a dark streak.

5. **Sharpness of streaks**

The last feature is the sharpness of a streak. Sharpness is defined by the boundaries between zones of different tones or colors. One way to measure the sharpness is to use the rise distance of the edge, for example, the distance (in pixels, millimeters, or fraction of image height) for the pixel level to go from 10% *to* 90% of its final value. This sharpness calculation method is called the $10-90\%$ rise distance. Firstly, we should project the ΔE value of each streak detection. Then, using the Eq. (6) to calculate the sharpness of streak, we define the S_{streak} to be the sharpness of streak. Figure 6 shows the notation of the Eq. (6).

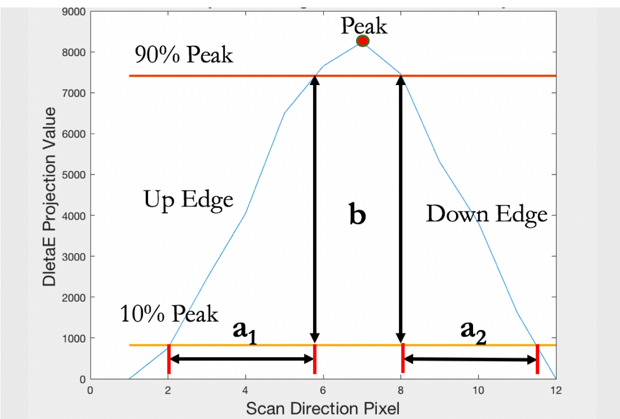


Figure 6: Calculation sharpness using the ΔE projection of one streak.

$$S_{streak} = \frac{2 \times b}{(a_1 + a_2)} \tag{6}$$

After extracting the features, we should normalize every feature for the Logistic Regression classification.

2.2.2 The data set

The streaks data set includes 24 streak defect images with 1699 pre-detected streak results. These 24 defect images include four different kinds of streak defects. Table 1 shows the details of the data set.

We use twenty images for the training set and four images for the test set.

Table 1: Streak Images Data Set

Defect Type	Number of Images
Developer Streak	10
ITB Streak	7
OPC Streak	11
Scanner Dirty Streak	3

2.2.3 Construct the logistic regression model

We assume this classification data needs a linear decision boundary. Eq. (7) shows the boundary function [15].

$$\theta_0 + \theta_1 x_1 + ... + \theta_n x_n = \sum_{i=0}^{n} \theta_i x_i = \theta^\top x \tag{7}$$

Based on the linear decision boundary assumption, we can build a the logistic regression prediction function Eq. (8).

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^\top x}} \tag{8}$$

In the Eq. (8), the $h_\theta(x)$ means "the probability of sample belongs to class 1". So, the probability of output belongs to class 1 or 0 is Eq. (9):

$$P(y=1|x;\theta) = h_\theta(x) \quad , \quad P(y=0|x;\theta) = 1 - h_\theta(x) \tag{9}$$

We can write the final predict function as Eq. (10).

$$P(y|x;\theta) = (h_\theta(x))^y (1 - h_\theta(x))^{1-y} \tag{10}$$

For m samples condition, the likelihood function is Eq. (11).

$$L(\theta) = \prod_{i=1}^{m} P(y^{(i)}|x^{(i)};\theta) = \prod_{i=1}^{m} (h_\theta(x^{(i)}))^{y^{(i)}} (1 - h_\theta(x^{(i)}))^{1-y^{(i)}} \tag{11}$$

We can write the log likelihood function Eq. (12).

$$l(\theta) = \log L(\theta) = \sum_{i=1}^{m} (y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))) \tag{12}$$

We can construct the final cost function Eq. (14) based on Eq. (13).

$$J(\theta) = -\frac{1}{m} l(\theta) \tag{13}$$

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^{m} y^{(i)} \log h_\theta((x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) \right] \tag{14}$$

Using the gradient descent method Eq. (15) with step size α to calculate the minimum of the cost function, we can get the best classification parameter θ, until the difference value between two neighboring iterations cost function satisfies Eq. (16). Figure 7 shows the cost value as a function of iteration number.

$$\theta_{j+1} = \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j} \quad , \quad (j = 0, 1, ...n) \tag{15}$$

$$\theta_{j+1} = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)} - y^{(i)})) x_j^{(i)} \quad , \quad (j = 0, 1, ...n)$$

$$|J(\theta_j) - J(\theta_{j-1})| < 0.0001 \tag{16}$$

2.2.4 Prediction result and cross validation

Figure 8 shows the prediction result. Two more predict sample results are shown in Figure 9 and Figure 10.

In the logistic regression process, we use the K-fold cross-validation to examine the model. The original samples with 22 images is randomly partitioned into 11 equal sized subsamples. For each subsample, we take it as a test data set, and the remaining subsamples as a training data set. Then, we fit the logistic regression model on the training set and evaluate it on the test set. After 11 iterations, we can calculate the confusion matrix for the logistic regression algorithm. Table 2 shows the confusion matrix, and Eq. (17) shows the accuracy of the logistic regression prediction result.

$$Accuracy = \frac{TP+TN}{Total} = \frac{1541+147}{1699} = 99.35\% \qquad (17)$$

Table 2: Confusion Matrix of Logistic Regression

	Predict: Invisible	Predict: Visible
Groundtruth: Invisible	1541	0
Groundtruth: Visible	11	147

3. Conclusion

In this paper, we propose a method to detect different kinds of streak in a test image. This method includes two techniques: a Block Window Method and the Logistic Regression Algorithm. The results indicate that the prediction performance of logistic regression is generally good. The accuracy is about 99.35%. That means that the logistic regression method can remove most noise detections.

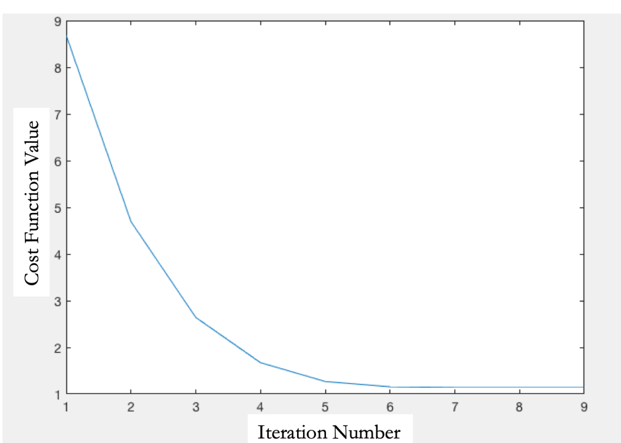


Figure 7: The cost value as a function of iteration number.

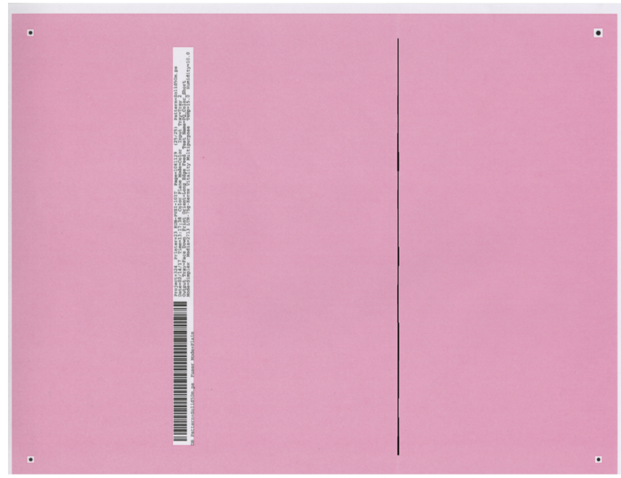


Figure 8: The final detection result.

References

[1] Z. Xiao, M. Nguyen, E. Maggard, M. Shaw, J. Allebach, and A. Reibman, “Real-time print quality diagnostics,” in Electronic Imaging, Color Imaging XIV: Image Quality and System Performance, Burlingame, CA, January 2017.

[2] Z. Xiao, S. Xu, E. Maggard, K. Morse, M. Shaw and J. Allebach, “Detection of color fading in printed customer content” in Electronic Imaging, Color Imaging XV: Image Quality and System Performance, Burlingame, CA, January 2018.

[3] M. Q. Nguyen, S. Astling, R. Jessome, E. Maggard, T . Nelson, M . Q. Shaw, and J. P. Allebach, “Perceptual Metrics and Visualization Tools for Evaluation of Page Uniformity,” Image Quality and System Performance XI, SPIE Vol. 9016, S. Triantaphillidou and M.-C. Larabi, Eds. San Francisco, CA, 3-5 February 2014.

[4] W. Wang, G. Overall, T. Riggs, R. Silveston-Keith, J. Whitney, G. T. C. Chiu, and J. P. Allebach, “Figure of Merit for Macrouniformity Based on Image Quality Ruler Evaluation and Machine Learning Framework,” Image Quality and System Performance X, SPIE Vol. 8653, P. D. Burns and S. Triantaphillidou, Eds. San Francisco, CA, 3-7 February 2013.

[5] X. Liu, G. Overall, T. Riggs, R. Silveston-Keith, J. Whitney, G. T. C. Chiu, and J. P. Allebach, “Wavelet-Based Figure of Merit for Macrouniformity,” Image Quality and System Performance X, SPIE Vol. 8653, P. D. Burns and S. Triantaphillidou, Eds. San Francisco, CA, 3-7 February 2013.

[6] M. Q. Nguyen and J. P. Allebach, “Controlling Misses and False Alarms in a Machine Learning Framework,” Image Quality and System Performance XII, SPIE Vol. 9396, M.-C. Larabi and S. Triantaphillidou, Eds. San Francisco, CA, 8-12 February 2015.

[7] X. Jing, S. Astling,R . Jessome, E. Maggard, T. Nelson, M. Q. Shaw, and J. P. Allebach, “A General Approach for Assessment of Print Quality,” Image Quality and System Performance X, SPIE Vol. 8653, P. D. Burns and S. Triantaphillidou, Eds. San Francisco, CA, 3-7 February 2013.

[8] X. Jing, H. Nachlieli, D. Shaked, S. Shiffman, and J. P. Allebach, "Masking Mediated Print Defect Visibility Predictor," Image Quality and System Performance IX, SPIE Vol. 8293, F. Gaykema and P. D. Burns, Eds, San Francisco, CA, 23-26 January 2012.

[9] J. Zhang, S. Astling, R. Jessome, E. Maggard, T. Nelson, M. Q. Shaw, and J. P. Allebach, "Assessment of Presence of Isolated Periodic and Aperiodic Bands in Laser Electrophotographic Printer Output," Image Quality and System Performance X, SPIE Vol. 8653, P. D. Burns and S. Triantaphillidou, Eds. San Francisco, CA, 3-7 February 2013.

[10] S. Hu, H. Nachlieli, D. Shaked, S. Shiffman, and J. P. Allebach, "Color-Dependent Banding Characterization and Simulation on Natural Document Images," Color Imaging XVII: Displaying, Processing, Hardcopy, and Applications, SPIE Vol. 8292, R. Eschbach, G. Marcu, and A. Rizzi, Eds., San Francisco, CA, 23-26 January 2012.

[11] J. Zhang, H. Nachlieli, D. Shaked, S. Shiffman, and J. P. Allebach, "Psychophysical Evaluation of Banding Visibility in the Presence of Print Content," Image Quality and System Performance IX, SPIE Vol. 8293, F. Gaykema and P. D. Burns, Eds, San Francisco, CA, 23-26 January 2012.

[12] W. Wang, Y. Guo, and J. P. Allebach, "Image Quality Evaluation Using Image Quality Ruler and Graphical Model," Proceedings of ICIP-2015 IEEE International Conference on Image Processing, Quebec City, Canada, 27-30 September 2015

[13] C. Lu, B. Shen, L. Zhang, J. P. Allebach, "Tag Recommendation via Robust Probabilistic Discriminative Matrix Factorization," IEEE International Conference on Acoustics, Speech and Signal Processing 2016, ICASSP Page 1170-1174, Z. Ding, W. Zhang, Z. Luo, Shanghai, China, 20-25 March 2016.

[14] C. Lu, J. Wagner, B. Pitta, D. Larson, J. P. Allebach, "SVM-based Automatic Scanned Image Classification with Quick Decision Capability," Proc.SPIE, Vol. 9015, 2014.

[15] D.W. Hosmer, Jr, S. Lemeshow, S. X. Sturdivant, "Applied Logistic Regression," Vol. 398, John Wiley & Sons, Inc., Hoboken, New Jersey, 2013.

Author Biography

Runzhe Zhang received his B.S. degree in Mechanical Engineering from Qingdao University of Technology (2013), Shandong, China and received the M.S. degree in Mechanical Engineering from Boston University, MA, USA. Currently, he is pursuing Ph.D. in Electrical and Computer Engineering at Purdue University. His research areas include digital image processing, computer vision, and machine learning.

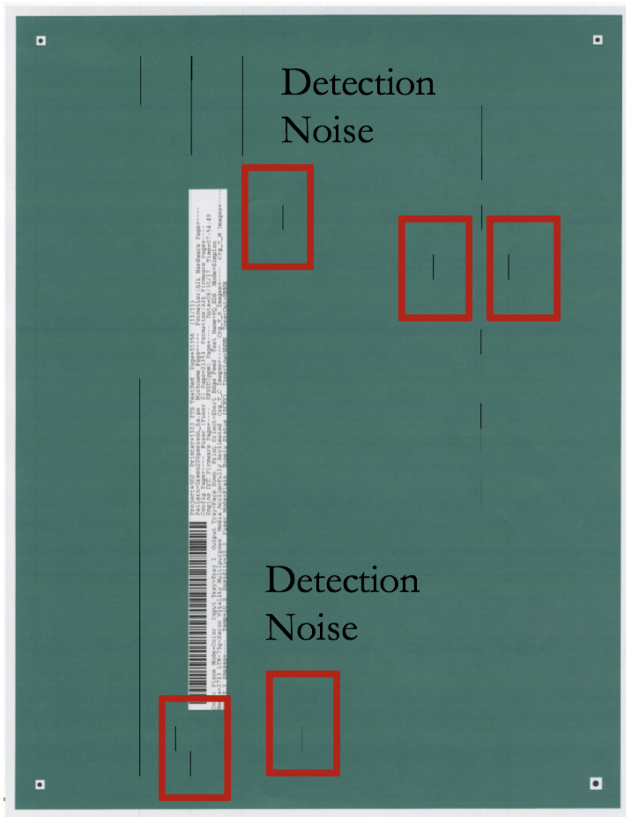
Eric Maggard received his B.S. degree in Physics from Northwest Nazarene University, Nampa, Idaho in 1991 and the M.S. degree in Computer Science specializing in image analysis and processing from Walden University in 2006. He is an Expert Imaging Scientist in the LaserJet Hardware Division and has developed programs and image quality algorithms for the last 15 years that are used in the testing of LaserJet print and scan image quality. His interests include machine vision, object recognition, machine learning, robot control, and navigation.

Renee Jessome received a B.S. in Mechanical Engineering from the Rochester Institute of Technology, Rochester, NY, in 1992. She has worked on a variety of printing devices over the past 14 years from the smallest LaserJet printers to Indigo Printing presses. She currently leads a Product Development team focused on high-end LaserJet Printers and MultiFunction Printers at HP Inc. in Boise, Idaho. During her early career at HP, she was developing storage devices such as disk drives, tape drives, and disk arrays.

Yousun Bang is a manager of Image Quality Part in Imaging Lab at HP Printing Korea Co. Ltd. She received her BS and MS in mathematics from Ewha Womans University, Seoul, Korea in 1994 and 1996, and her PhD in the school of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana in 2005. She worked for Samsung Advanced Institute of Technology and Samsung Electronics Company from 2004 to 2017. Her current research interests include image quality diagnosis and metrics and ML/DL based prediction for smart device services.

Minki Cho is the engineer of HP Printing Korea. He received B.S.(1997) and M.S.(1999) in electrical engineering from Inha University, Korea. From 2003 to 2017, he worked for Samsung Electronics and Samsung Advanced Institute of Technology. His research areas are print image processing, print image quality diagnosis & calibration. Recently, he is researching above interesting using ML & DL.

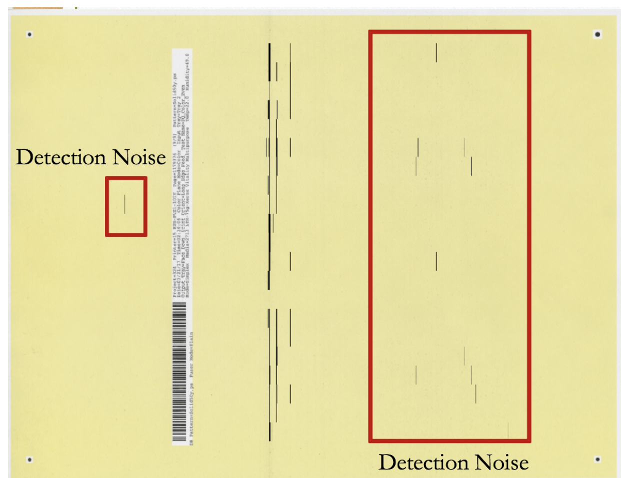
Jan P. Allebach is Hewlett-Packard Distinguished Professor of Electrical and Computer Engineering at Purdue University. Jan P. Allebach is a Fellow of the IEEE, the National Academy of Inventors, the Society for Imaging Science and Technology (IS&T), and SPIE. He was named Electronic Imaging Scientist of the Year by IS&T and SPIE and was named Honorary Member of IS&T, the highest award that IS&T bestows. He has received the IEEE Daniel E. Noble Award and the IS&T/OSA Edwin Land Medal and is a member of the National Academy of Engineering.



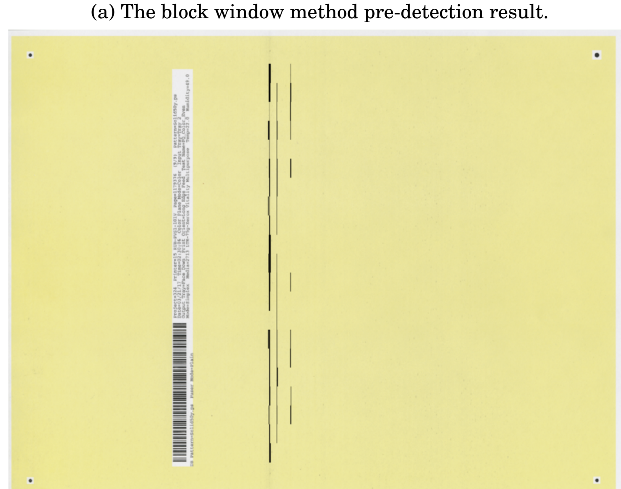
(a) The block window method pre-detection result.

(b) The streak detection result after logistic regression method.

Figure 9: The sample for block window method and logistic regression result.



(a) The block window method pre-detection result.



(b) The streak detection result after logistic regression method.

Figure 10: The sample for block window method and logistic regression result.