

Characterizing Streaks in Printed Images: A Matching Pursuit Method using Wavelet Decomposition

Juan Liu ¹, Wencheng Wu ², Bob Price ¹, Eric Hamby ², and Raj Minhas ²

¹ Palo Alto Research Center, Palo Alto, CA

{jjliu, bprice}@parc.com

² Xerox Research Center at Webster, Webster, NY

{wencheng.wu, eric.hamby, raj.minhas}@xerox.com

ABSTRACT

Printed pages from industrial printers can exhibit a number of defects. One of the common defects and a key driver of maintenance costs is the line streak. This paper describes an efficient streak characterization method for automatically interpreting scanned images using the matching pursuit algorithm. This method progressively finds dominant streaks in signal profiles. It uses wavelet decomposition to speed up the element selection process and reduce computation complexity. Previous approaches require the design engineer to pre-specify the characteristics for each possible streak that could be detected – an approach which is practically limited to detecting a few streak types in specific locations. The Matching Pursuit algorithm, in contrast, fully characterizes any and all streaks found on the scanned page permitting a generic analysis of a broad range of defects found in the field.

1. INTRODUCTION

Printers are highly complex electro-mechanical systems. Hundreds of components are involved in the printing process, which in turn has multiple stages: charging, exposure, development, transfer, fusing, and cleaning (Duke, Noolandi, & Thieret, 2002). Printers are designed with remarkable reliability with failure rates measured in incidents per millions of impressions; however, modern printing applications print huge volumes of sheets so diagnosis remains an important step in maintaining asset availability. Diagnosing a printer to isolate a defective component is often a difficult task due to the printer's formidable complexity. Experienced technicians learn to recognize signature defects in order to quickly narrow down the responsible components. In our project, we aim to automate defect diagnosis. Automation would allow the printer to suggest required parts before the technician departs for the site, speed up the diagnostic process for less experienced technicians and allow

This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

more sophisticated and demanding customers to take on more of the maintenance tasks on site and without the delays associated with service calls.

We propose the use of iterative diagnosis, in which three steps are interleaved and performed in each iteration:

1. printed pages are scanned and their image quality features are computed and characterized;
2. image quality features are used to update the health state of the machine and refine diagnostic hypotheses;
3. the set of hypotheses is used to recommend further tests to rule out ambiguities or repair actions to correct problems found so far

The process repeats until the problem with the machine has been diagnosed.

There is a body of literature on fault diagnosis (the second step in iterative diagnosis). Various techniques ranging from qualitative to statistical inference have been proposed. The third step in iterative diagnosis is essentially evidence seeking. A few information-driven approaches (e.g., (Liu et al., 2008), (de Kleer & Williams, 1987)) have been suggested to find informative evidence. We will not address these two problems in this paper. Instead, we focus on the first step of the iterative diagnosis process, namely characterization of image quality problems. This can be considered as a pre-processing stage, or equivalently feature extraction. Printout data are distilled into concise features, which are then fed to a diagnoser that classifies faults.

Banding and streaking are two typical image quality problems (Rasmussen, Dalal, & Hoffman, 2001). Banding is often caused by defects in rotational components such as the rotating mirror used to raster-scan an image. It exhibits periodicity along the process direction of the paper. This periodicity makes it easy to detect and characterize. Since the periodicity correlates with the circumference of the part causing the banding, diagnosis is straight forward. On the other hand, streaks appear as stripes whose major axis is along the process direction and are caused by defects or degradation in a variety of components related to the xerographic process. Streaks typically do not exhibit periodicity and vary considerably

in appearance. In this paper, we focus on the complexities of streak characterization.

Streak characterization has its own value even when detached from fault diagnosis. Currently, experienced technicians describe streaks using descriptions such as “light narrow streaks” or “dark medium streaks”. This is highly subjective, varying with individual perception, and hence inaccurate. The algorithm that we propose eliminates the subjective nature of streak description and can be used as an automated tool for assisting technicians in assessing image quality problems and providing objective measures of whether or not a printer is meeting performance standards.

The objective of this work is to develop an algorithm that automatically characterizes streaks in printed pages. Such an algorithm needs to be:

- intuitive: the algorithm produces a representation of streaks consistent with artifacts identified by human operators;
- concise: streak characterization is clear and succinct.
- flexible: the algorithm can successfully characterize a wide variety of streaks.

Prior work in this area has focused on streak detection (Rasmussen et al., 2001). For instance, if the systems engineer anticipates that paper running through the system could create media edge wear on a roller at a particular spot, the engineer could specify a specific filter that would fire when a streak appears in the expected location with the expected width and polarity. This approach is often implemented using matched filters, in which a filterbank is built based on the specified streak characteristics. The detection is straightforward and reliable. However, it is limited in flexibility because it requires pre-specification of streak patterns. Unfortunately, it is impractical for engineers to prespecify specific filters for all streaks observed in the field. In this paper, we seek to broaden the scope from streak detection to streak characterization, in the sense that we describe features of any and all prominent streaks that can be observed on a printed sheet. Increased flexibility comes at the cost of increased computation complexity. However, we show that discrete wavelet transform like techniques can make these algorithms tractable for use in real systems.

2. EXAMPLE STREAKS

Figure 1 shows a few representative streaks. As shown in the figure, streaks vary in appearance. A streak could be narrow or wide, light or dark. It may occur at a fixed location over different pages, or occur at random locations. A streak defect could take the form of a single isolated streak (Figure 1 a, b, and d), or it could be a cluster of streaks (Figure 1 c).

Failure in various printer components can cause streaks. For instance, the streaks in Figure 1 a and d are caused by defects in the charging process (creating a high voltage field on a photoreceptor which is sculpted by the laser to make an electrostatic image). They tend to be fuzzy and occur in clusters. The streaks in Figure 1c are due to defects in the cleaner blade (which scrapes residual toner off the drum once the image has been transferred). They tend to be sharper edged and extend outside the printable area. Other defects such as the

failure to properly develop (distribute toner based on the electrostatic image) or to transfer (move toner from the machine to paper) result in streaks with specific characteristics. Being able to precisely quantify such characteristics is crucial to diagnosis.

3. FORMULATION AS A SIGNAL DECOMPOSITION PROBLEM

Ideally, a streak characterization algorithm would start with a customer image. Due to the highly variable nature of these images, it is difficult to accurately identify all streak types in such an image. We therefore assume that the operator can print a preprogrammed test sheet with homogeneous half-toned color test patches. Since test patches are homogeneous, and streaks generally run across the entire patch in the process direction, we can project the streak along the process direction via integration or averaging, and reduce printed images to 1-D signal (see (Rosario, Saber, Wu, & Chandu, 2007)). The collapsed 1-D signal is known as the “density profile”¹ and captures variation along the cross-process direction. In some cases, streaks can vary along their length. To accurately isolate features related to this variability would require analyzing the original unprojected 2-D intensity field. We defer this to future work.

Key characteristics of streaks, include location, width, intensity, length, and edge sharpness. In this work, we capture a subset of these properties in terms of a simple mathematical parameterization: (τ, α, h) , where τ denotes location, α denotes scale (inversely proportional to streak width), and h denotes intensity. The goal is to extract a series of descriptor tuples $\{(\tau_i, \alpha_i, h_i)\}_{i=1,2,\dots,N}$ from an 1-D profile $f(t)$.

We formulate the problem as the following: given the 1-D profile $f(t)$, we would like to decompose it as a series of superimposed streaks:

$$f(t) = c + \sum_i h_i x(\alpha_i t - \tau_i), \quad (1)$$

where c is a constant corresponding to the average intensity of the solid color test patch, and $x(t)$ is streak template (e.g., raised cosine or block-wave function). The streak template $x(t)$ is stretched or squeezed to proper width by the scale parameter α_i , and shifted to location τ_i . Its intensity is modified by the intensity or height parameter h_i . We would like to seek tuples (τ_i, α_i, h_i) such that the summation on the right-hand side of (1) matches with the observation profile $f(t)$. Essentially, we have a signal decomposition problem, where $\{x(\alpha_i t - \tau_i)\}$ is a set of basis functions, onto which the signal $f(t)$ can be projected. In practice, the profile $f(t)$ is often contaminated by noise and other printing and scanning artifacts. Hence, we would like to seek $\{(\tau_i, \alpha_i, h_i)\}_{i=1,2,\dots,N}$ to minimize the discrepancy

$$\mathcal{E} = \int_{t \in \mathbb{R}} |f(t) - c - \sum_i h_i x(\alpha_i t - \tau_i)|^2 dt. \quad (2)$$

The advantage of the signal decomposition approach is that it is capable of describing complicated streak artifacts, for instance, two narrow streaks on top of a wide

¹We use the convention that 1-D signal refers to a signal of a single variable, e.g., $f(t)$, and 2-D signal/image/field to refer to a signal of two variables, e.g., $f(x, y)$.

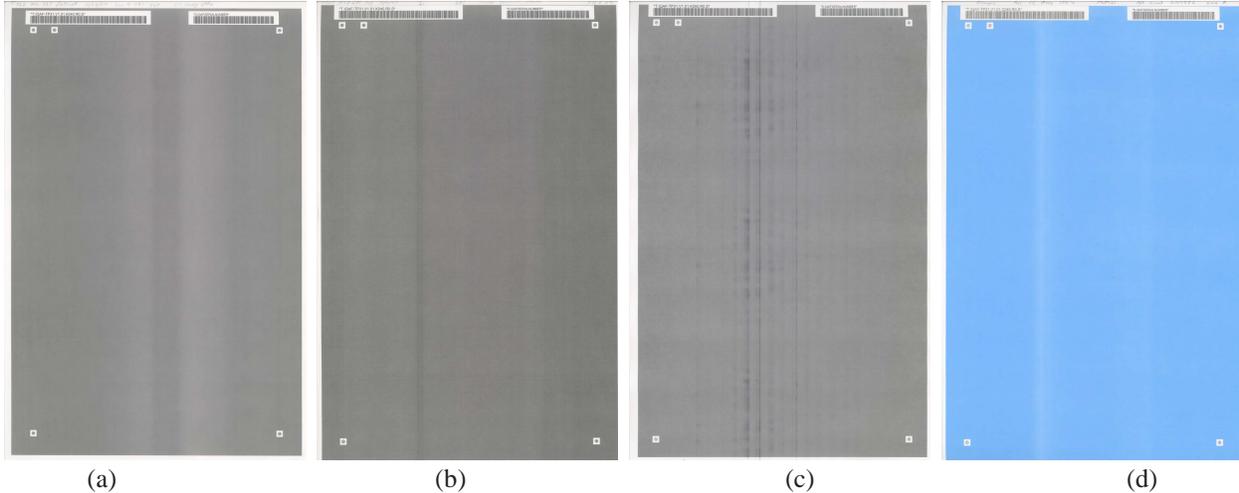


Figure 1: Example streaks. From left to right — (a) light and dark wide streaks; (b) dark streak with medium width; (c) clustered dark narrow streaks; (d) light wide streak.

streak. This can be easily expressed as the superimposition of three streaks, two with large α_i values, and one with small α_i . Another advantage is its remarkable flexibility. It only requires the engineer to specify a single generic streak template $x(t)$. The algorithm looks for the optimal set of instances $\{(\tau_i, \alpha_i, h_i)\}_{i=1,2,\dots,N}$. There is no need for engineer to specify detailed streak information and pre-construction of any filterbanks.

This signal decomposition approach also has drawbacks. The foremost problem is the non-uniqueness in representation: the space of possible basis functions $\{x(\alpha t - \tau), \tau \in \mathbb{R}, \alpha \in \mathbb{R}^+\}$ is clearly over-complete, (i.e., one member can be expressed as the linear combination of other members). To make the difficulty clear, assume for a moment that we are using a block-wave streak template. A block-wave function of width 1 can be expressed as the sum of two adjacent block-wave functions each with width 1/2, and likewise four block-wave functions with width 1/4, and so on. This inherent over-completeness will result in non-uniqueness in the signal decomposition $\{(\tau_i, \alpha_i, h_i)\}_{i=1,2,\dots,N}$. In theory, there exist an infinite number of signal decompositions with equal signal representation error \mathcal{E} . In our approach, we use a matching pursuit method, described in Sec. 4., to look for the most concise signal decomposition which circumvents the uniqueness problem.

Another difficulty with the signal representation approach is the computation complexity. In principle, we compute the projection of $f(t)$ onto the basis set $\{x(\alpha t - \tau), \tau \in \mathbb{R}, \alpha \in \mathbb{R}^+\}$. This involves searching for the best decomposition over $\tau \in \mathbb{R}, \alpha \in \mathbb{R}^+$. In practice, our signal is sampled at finitely many points on a finite extent so there will not be an infinite number of decompositions, but there will still be an intractably large set. Our solution to this problem is to use the discrete wavelet transform (DWT) to speed up the search. The use of DWT in signal decomposition is similar to our signal decomposition problem in the sense that it projects a signal onto a set of basis functions, but the basis set in DWT is complete and structured (defined by the wavelet function). Assuming that wavelet function

is similar to streak template, DWT could be used to approximately match locations and scales. This truncates the search domain of τ and α from the entire plane (or half plane) to much smaller neighborhood which can be searched directly. Furthermore, the DWT computation can be implemented using efficient and readily available algorithms. The DWT acceleration of matching pursuit is described in more details in Sec. 5..

4. MATCHING PURSUIT

The previous section explains that signal decomposition onto the streak template basis is inherently non-unique. Given that multiple decompositions are equally good in representing the original signal $f(t)$, intuitively one would like to favor the decomposition which is most sparse, i.e., with the least number of basis functions. This is well-aligned with the principle of parsimony: the simplest explanation should be favored.

Matching pursuit is an efficient and intuitive method, originally proposed by Mallat and Zhang (Mallat & Zhang, 1993) for basis selection. It has the notion of a dictionary, which is a collection of waveforms that can be used to describe a signal $f(t)$. For instance, the Fourier basis is good at describing periodicity, and the wavelet basis is good for describing locality. The dictionary may contain one of the two, both, or even more. The task is to select members from the dictionary in order to best describe $f(t)$. This is very similar to our streak characterization problem, where the dictionary is $\{x(\alpha t - \tau), \tau \in \mathbb{R}, \alpha \in \mathbb{R}^+\}$. The dictionary is redundant, and hence the same non-uniqueness problem needs to be addressed. Matching pursuit proposes a greedy strategy: it progressively builds up a signal representation by selecting an element that maximally improves the representation accuracy in each iteration.

Matching pursuit starts with the original signal $f(t)$, and finds the element $g(t)$ in the dictionary which best matches with $f(t)$. Given a template $g(t)$, the best ap-

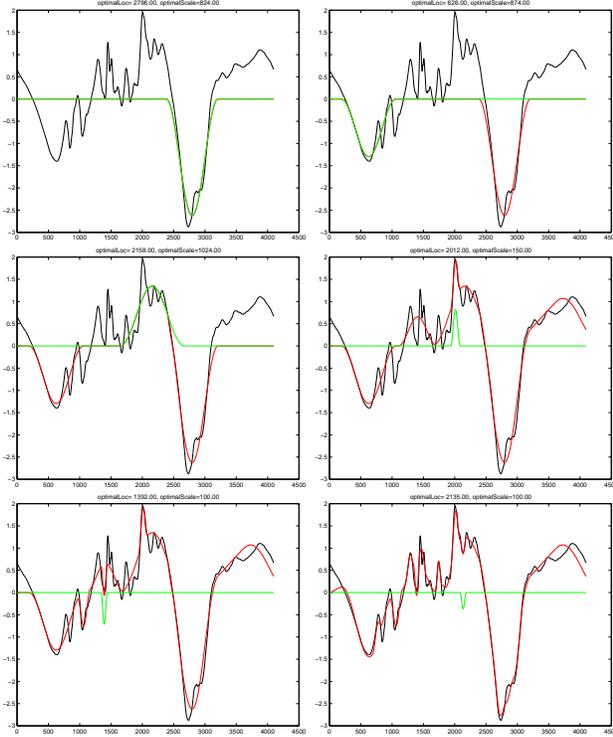


Figure 2: Progression of matching pursuit: top row — the 1st and 2nd iterations of matching pursuit; middle row — the 3th and 7th iteration; bottom row — the 9th and 30th iteration. The black curves are the 1-D profile signal $f(t)$. The green curves are the best template matched signal ($f^{proj}(t)$) in iterations. The red curves are reconstructed signal by the end of each iteration.

proximation can be defined as

$$f^{proj}(t) = \langle f(t), \frac{g(t)}{\|g(t)\|} \rangle \cdot \frac{g(t)}{\|g(t)\|}. \quad (3)$$

Here the notation $\langle f, g \rangle$ stands for inner product; and $\|g\|$ denotes Euclidean norm. The matching error is $\mathcal{E}_g = \|f - f^{proj}(t)\|$. We search over all $g(t)$ to minimize the matching error. The signal $f(t)$ is updated by the approximation residual, i.e.,

$$\begin{aligned} f(t) &\leftarrow f(t) - f^{proj}(t) \\ &= f(t) - \langle f(t), \frac{g(t)}{\|g(t)\|} \rangle \cdot \frac{g(t)}{\|g(t)\|} \end{aligned}$$

Then the matching pursuit iterates and looks for the next best match. The iteration stops when a maximal number of iterations is reached, or when the remainder signal $f(t)$ contains very little energy. The intuition behind matching pursuit is that by progressively identifying the most dominant match, the signal representation will be sparse. While there is no guarantee of sparsity, matching pursuit can successfully identify dominating basis, is easily implemented, converges quickly, and produces accurate representation.

Figure 2 illustrates the progression of matching pursuit, showing the optimal matched templates in iterations

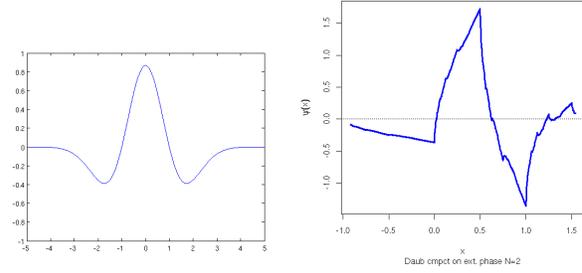


Figure 3: Mexian hat (left) and Daubechies wavelet (right).

1, 2, 3, 7, 9, and 30. The original curve is shown in black. The green curve represents a greedy match of the best single half-cosine basis function to the original curve. The red curve shows the signal reconstruction in terms of the basis curves identified so far. As we get to later iterations, the red reconstruction curve comes to approximate the original black signal and the identified streaks are increasingly weak in terms of the energy they capture. This matching pursuit method is capable in characterizing a variety of streak, wide (in the top row) or narrow (in the bottom row). It allows us to describe artifacts such as overlapping streaks.

While matching pursuit works efficiently, it does not guarantee sparsity. There is a body of literature on the optimal tradeoff between sparsity and representation accuracy. Several heuristic methods have been developed to improve matching pursuit. Interested readers may refer to work on basis pursuit (Chen & Donoho, 1994; Huggins & Zucker, 2007) for more elaborate techniques.

5. USING WAVELET DECOMPOSITION TO SPEED UP

Matching pursuit is inherently computation-intensive. In each iteration, matching pursuit finds the best basis function in the dictionary. At each step it must scan the whole dictionary. In our case, the directory is parametrized by the discretized location τ and scale α parameters. For each element (τ, α) , computing the projection of $f(t)$ onto $x(\alpha t - \tau)$ is $O(N)$ where N is the number of samples in $f(t)$. Therefore, the overall complexity for each iteration is $O(N \cdot |\tau| \cdot |\alpha|)$.

We can accelerate matching pursuit by using the discrete wavelet decomposition (DWT). Wavelet decomposition computes the projection of $f(t)$ onto a wavelet function $\psi(2^k(t - \tau))$, where t and τ are discrete, and k is the decomposition level. The decomposition level k corresponds to a scale or width of 2^k . Two facts justify the choice of DWT as an approximation to half-cosine matching:

- The wavelet function $\psi(t)$ is visually similar to streak template $x(t)$. For instance, Figure 3 plots the Mexican hat and Daubechies wavelet. Mexican hat is similar to a raised cosine streak template, and the dominant peak in Daubechies wavelet is similar to a triangle template.
- DWT has fast algorithms. DWT is computed by convolving $f(t)$ with filterbanks at dyadic scales.

Listing 1: Matching pursuit iterations

```

xRec=0;
matchInfo= []; %matchInfo contains tuple [loc,scale,intensity]
for iter= 1:maxIterNum
    % compute the best match
    % returns the projection signal optimalTemp, best match location, and scale
    [optimalTemp, bestMatchLoc, optimalScale]= findBestTemplateMatch(XSignal);

    % add the matched template to the reconstruction signal
    xRec= xRec+ optimalTemp;
    % record the descriptor (loc,scale,intensity)
    matchInfo= [matchInfo; bestMatchLoc optimalScale optimalTemp(bestMatchLoc)];

    % subtract the matched template from the original signal
    XSignal= XSignal- optimalTemp(:);

    if (norm(XSignal)> energyThresh) % very little energy left
        break;
    end
end
return (matchInfo);

```

Convolution can be efficiently implemented via fast Fourier transform (FFT).

Peaks and valleys of DWT at decomposition level k and location τ indicates that there is a good match between $f(t)$ and the wavelet basis $\psi(2^k(t - \tau))$. Given the similarity between the wavelet function $\psi(t)$ and streak template $x(t)$, it is also reasonable to assume that the match is also good between $f(t)$ and $x(2^k(t - \tau))$. Hence DWT is indicative of potential streak locations and scales. Rather than searching through all the possible elements in the (τ, α) domain, we can use DWT to speed up for the optimal basis search: (1) first identifying potential match locations and scales, and (2) restricting the search to a much smaller neighborhood.

Furthermore, DWT also works well for preprocessing, including denoising and baseline removal. DWT is suitable for denoising due to its energy compaction property — energy in smooth signal is compacted into only a few significant coefficients, while noise energy is widely scattered. Baseline removal comes in for free because the coarsest subband naturally provides a low-pass approximation. We will discuss these in more details in the algorithm section.

Peak detection using continuous wavelet transform has been proposed in bioinformatics application in (Du, Kibbe, & Lin, 2006). The basic idea is the same, but it does not enjoy the fast computation as DWT does.

6. ALGORITHM

6.1 From images to 1-D profiles

As we have discussed in Sec. 3., we first reduce 2-D images to 1-D profiles for streak characterization. This is a non-trivial preparation step. In practice, when the printed page is scanned, it is often subject to mild distortion such as translation, rotation, and skewing. We have developed an image registration algorithm to correct such distortion. Each test page has fiducial marks (three on the top, two on the bottom in each page in

Figure 1). These fiducial marks are detected automatically. From the fiducial marks, the algorithm computes an affine transform, which transforms the scanned image to a standard coordinate frame where process direction is perfectly aligned with the vertical axis of the image coordinate. Once the image has been correct for distortions, one can easily compute the 1-D profile, simply via averaging the transformed image across the vertical direction.

6.2 Preprocessing

The purpose of preprocessing is to prepare data for matching pursuit. For instance, profile signals obtained from real-world images are often contaminated by non-streak noise in the printing/scanning process. Denoising is needed to reduce noise while preserving significant streaks. This can be done effectively using the DWT. Like Karhunen-Loeve transform, DWT compacts energy into just a few wavelet coefficients. In contrast, white noise affects all coefficients. We use a hard-thresholding scheme for denoising (Liu & Moulin, 1997): if a wavelet coefficient's amplitude is small with respect to a threshold, it is considered to be due to noise and is set to zero. Wavelet coefficients with large amplitude are considered signal and are preserved. In our implementation, the threshold is set to 3σ , where σ is the nominal standard deviation of noise, which is assumed known a priori, or can be learned from a set of observation samples. Figure 4a shows the raw 1-D profile (in black) and its denoised version (in red). Visually the red signal is much smoother but still preserves fine-level details.

In addition to streaks which appear as stripes, there is often slow variation from one margin to the other due to, for example, the uneven nature of ROS power in the cross-processing direction. This is known as the inboard-outboard variation, also referred to as the “baseline”. Removal of the baseline is necessary because it has a significant magnitude but the variation has different characteristics than streaks. The baseline is slowly varying

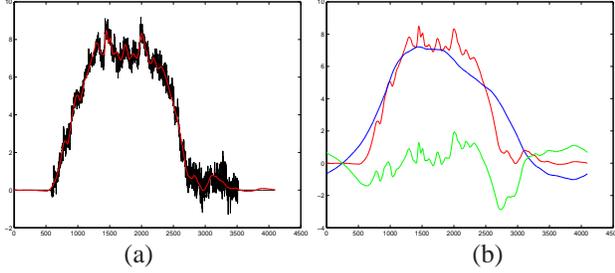


Figure 4: Removal of (a) non-streak scanning and printing noise and (b) inboard-outboard variation. Black is the raw 1-D profile; red is the denoised profile; blue is the identified baseline characterizing the inboard-outboard variation, and green is the profile after noise/baseline removal.

and has wide support (often the whole page). It is somewhat surprising that end users generally do not perceive the baseline as problematic, but it is spread out over a large enough area that it is not perceptually salient to humans. In contrast, streaks are also smooth, but have much smaller support (in the millimeter range). On the other hand, the difference between baseline and streaks are purely qualitative. It is hard to precisely define what part of the profile is due to baseline and which part is due to streaks. Perfect separation is impossible. This is an area that remains as our on-going research.

In our algorithm, with a N -level DWT, the signal is decomposed into $N + 1$ coarse-to-fine subbands. The coarsest subband is a low-pass version of $f(t)$, while the finer subbands are the projection of $f(t)$ onto the wavelet basis $\psi(2^k(t - \tau))$ for $k = 1, \dots, N$. The projections have a finite support proportional to 2^k . This subband structure provides a natural separation between baseline and streaks. In our implementation, we choose N so that 2^N is roughly the support of the widest streaks. For instance, the 1-D profile in Figure 4b is 4096 pixel long, we choose $N = 10$ which corresponds to streaks of width 1024. Anything wider than that is considered baseline instead of streaks. The baseline is obtained by reconstructing from the coarsest subband only. Figure 4b shows the denoised 1-D profile in red and the baseline in blue. The blue curve captures the general trend of inboard-outboard variation very well. The difference between the two is shown in green. It removes the low-frequency baseline from $f(t)$ but preserves the high-frequency variations. This is the input signal to the matching pursuit algorithm.

6.3 Combining DWT with matching pursuit

As we have discussed in Sec. 5., the purpose of DWT is to find potential streak candidates, in particular, the approximate scale and location (α, τ) , so that we can search over a small neighborhood instead of searching over the whole domain $(\alpha, \tau) \in \mathbb{R}^2$. This greatly saves computation time. Figure 5 shows the 1-D streak profile, from which we identify 5 candidates for the most dominant streaks. The candidate match locations are marked with squares. The candidate identification process is straightforward — the location with a large amplitude in the wavelet decomposition is identified as a po-

tential candidate. The figure shows that dominant streaks are located, but there are two problems: (1) the locations may not be accurate due to the fact that DWT uses discrete resolution (integer multiplier of 2^k , where k is the wavelet decomposition level) hence is incapable of precisely identifying location in between, and (2) the candidates could be repetitive, for instance, the top five candidates actually identifies three streaks in the profile. Both problems are not critical, since the matching pursuit algorithm will perform refined match to further improve location accuracy and remove redundancies.

The identified candidates $\{(\alpha_i, \tau_i)\}$ are then used to define search neighborhood for the matching pursuit algorithm. The bottom panel of Figure 5 shows the streak characterization result. Each streak is marked with a horizontal bar. The center location of the horizontal bar indicates the streak location τ_i ; the vertical location indicates the intensity of the streak h_i ; the length of the bar indicates the streak width $1/\alpha_i$. The top 5 streaks are labeled with numbers. From the figure we see that the streaks are identified correctly.

6.4 Finding correlated streaks across page or across color separations

In some cases, a single component may generate a streak in multiple separations. For instance, the fuser fuses the toner for all colors and therefore defects in the fuser roll create artifacts in all separations. To properly isolated the cause of these failures from image defects, it is necessary to compare streak characteristics between multiple pages.

The implication on streak characterization is that when it comes to streak detection over multiple channels (page or color), we should not treat each channel $f_j(t)$ separately. If $f_{j=1}(t)$ has a strong streak at (α, τ) , we should examine $f_{j=2, \dots, J}(t)$ in the (α, τ) neighborhood for potential streak presence. This has been implemented as the following in our algorithm:

- For each channel profile $f_j(t)$, use DWT to find streak candidate set $L_j = \{(\alpha, \tau)\}_j$;
- Generate an augmented signal as the Euclidean norm across all channels, i.e., $f_{aug}(t) = \|f_j(t)\|$, and then compute the candidate list $L_{aug} = \{(\alpha, \tau)\}_{j=J+1}$ for $f_{aug}(t)$;
- Augment each channel's candidate list as $L_j^{new} \leftarrow (L_j \cup L_{aug})$;
- For each channel profile $f_j(t)$, perform matching pursuit based on the augmented candidate list L_j^{new} .

The augmented channel f_{aug} is a new 2-D profile, which captures dominant streaks in any of the channels. Hence the candidate list L_{aug} identifies potential streaks across channels.

Figure 6 shows the streak characterization result for a test page with four color separations. The correlated streaks are located roughly 2/3 of the paper width from the left margin. The algorithm correctly identifies streaks in the cyan, black, and magenta separations.

7. FUTURE WORK

The initial work here demonstrates the potential of matching pursuit and discrete wavelet transform in char-

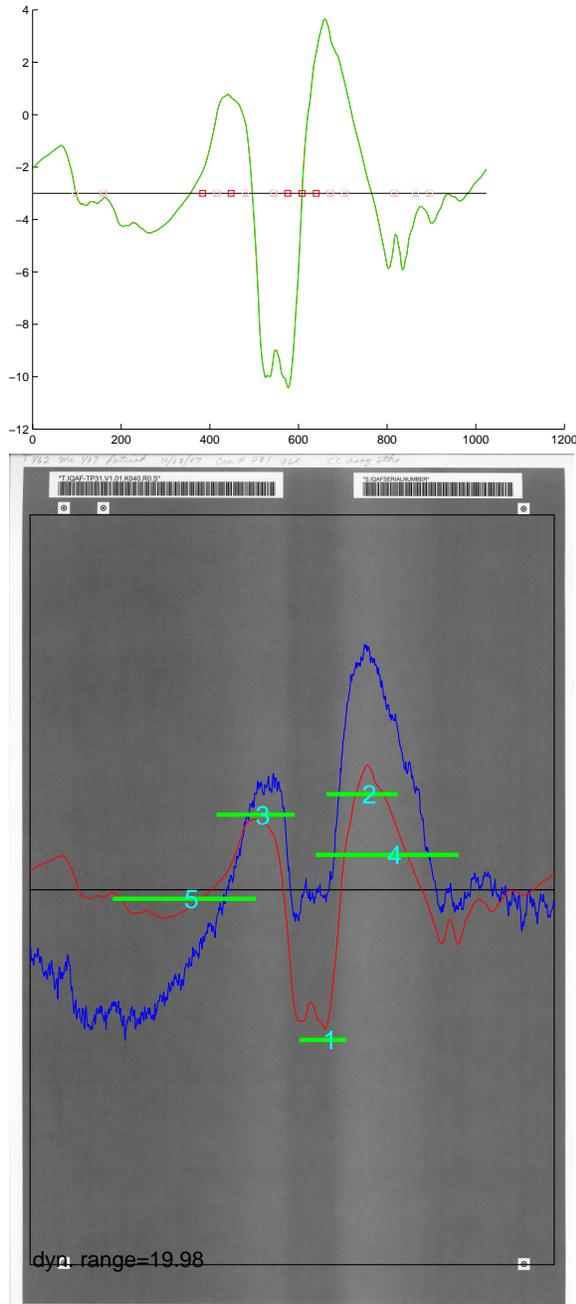


Figure 5: Top panel: red boxes show locations within 1-D profile identified as match candidates using DWT; bottom panel: streak characterization result overlaid on original scanned image

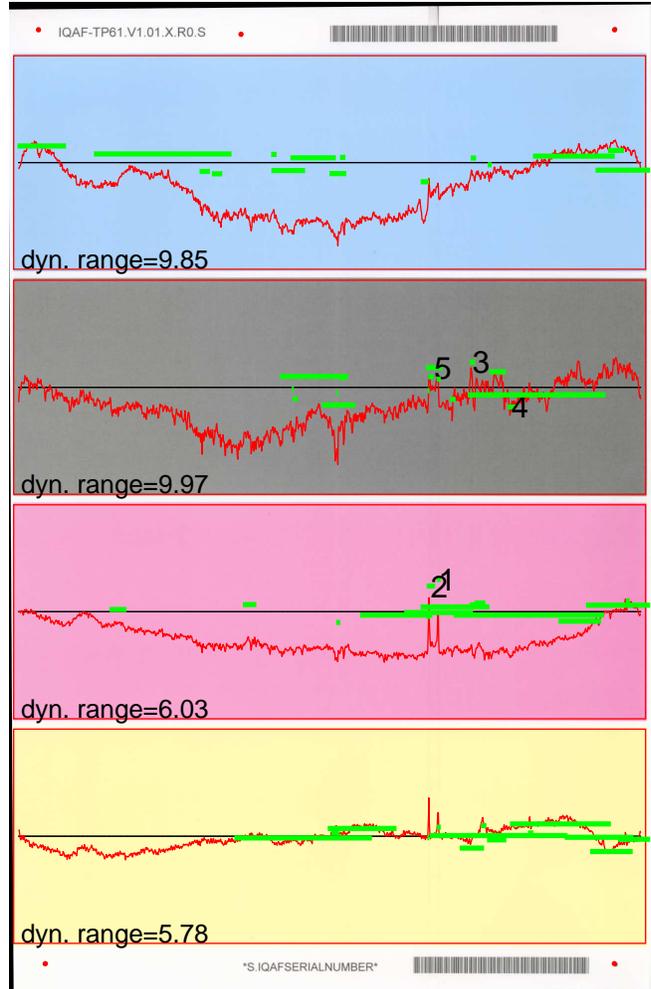


Figure 6: Example of a printed image with streaks (top patch, three strips due to scorotron fault). The 2-D profiles are plotted in red for all test patches.

acterizing streaks. The present method only returns location, scale and intensity. Extensions of the algorithm to identify additional features of streaks highlighted by subject matter experts as important for diagnosis represent natural starting points for future work. A more ambitious project would be the extension of the method to streaks which vary across the page. Such streaks would require methods that operate efficiently on 2-D intensity fields which we expect will be a challenge for some time to come.

8. CONCLUSIONS

The automated identification of streaks holds considerable promise for improving diagnosis and health management in printing systems but has proven difficult to formulate computationally. Prior methods have provided the ability to identify specific types of streaks in previously anticipated locations. In this paper we demonstrate a novel streak detector that dynamically identifies a broad family of streaks anywhere on the page. A key component to making this technology practical is to use

the discrete wavelet transform as an approximate matching heuristic to focus search for analytically intractable basis functions (half-cosines). The result is a robust and practical mechanism that will enable new levels image quality reliability throughout the printing industry.

REFERENCES

- Chen, S., & Donoho, D. (1994). Basis Pursuit. In *Proc. 28th Asilomar Conf. Signals, System Computation* (p. 41-44). Asilomar, CA.
- de Kleer, J., & Williams, B. C. (1987). Diagnosing Multiple Faults. *Artificial Intelligence*(32), 97-130.
- Du, P., Kibbe, W. A., & Lin, S. M. (2006). Improved Peak Detection in Mass Spectrum by Incorporating Continuous Wavelet Transform-Based Pattern Matching. *Bioinformatics*, 22(17).
- Duke, C. B., Noolandi, J., & Thieret, T. (2002). The surface science of xerography. *Surface Science*(500), 1005-1023.
- Huggins, P. S., & Zucker, S. W. (2007, July). Greedy Basis Pursuit. *IEEE Trans. Signal Processing*, 55(7), 3760-3772.
- Liu, J., Kleer, J. de, Kuhn, L., Price, B., Zhou, R., & Uckun, S. (2008). A Unified Information Criterion for Evaluating Probe and Test Selection. In *International Conference on Prognostics and Health Management (PHM)*.
- Liu, J., & Moulin, P. (1997, October). Complexity-regularized image denoising. In *Proc. of ICIP'97* (p. II. 370-373). Santa Barbara, CA.
- Mallat, S., & Zhang, Z. (1993, December). Matching Pursuits with Time-Frequency Dictionaries. *IEEE Trans. Signal Processing*, 41(12), 3397-3415.
- Rasmussen, D. R., Dalal, E. N., & Hoffman, K. M. (2001). Measurement of Macro-Uniformity: Streaks, Bands, Mottle and Chromatic Variations. In *IS&T's 2001 PICS Conference Proceedings*.
- Rosario, H. S., Saber, E., Wu, W., & Chandu, K. (2007). Streak detection in mottled and noisy images. *Journal of Electronic Imaging*, 16.