User-Assisted Ink-Bleed Reduction

Yi Huang, Michael S. Brown, Dong Xu

***Abstract*—This paper presents a novel user-assisted approach to reduce ink-bleed interference found in old manuscripts. The problem is addressed by first having the user provide simple examples of foreground ink, ink-bleed, and the manuscript's background. From this small amount of user-labeled data, likelihoods of each pixel being foreground, ink-bleed, or background are computed and used as the data-costs of a dual-layer Markov Random Field (MRF) that simultaneously labels all pixels in both the front and back side of the manuscript. This user-assisted approach produces better results than existing algorithms without the need for extensive parameter tuning or prior assumptions about the ink-bleed intensity characteristics.**

Our overall application framework is discussed along with details of the features used in the data-costs, a comparison between KNN and SVM for likelihood estimation, the dual-layer MRF formulation with associated inter- and intra-layer costs, and a comparison of our approach against other ink-bleed reduction algorithms.

***Index Terms*—Ink-bleed Reduction, Dual Layer MRF, Document Imaging Processing.**

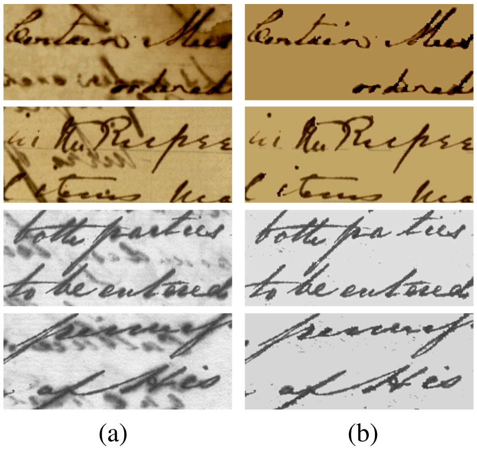


Fig. 1. (a) Closeup of image regions from different handwritten documents, circa 1820-1850, suffering from ink-bleed. (b) Our goal is to retain the original foreground strokes to improve legibility.

I. Introduction

The work presented in this paper is part of a collaborative effort with the National Archives of Singapore (NAS). The NAS houses several hundred volumes of handwritten manuscripts circa 1820-1860. These manuscripts are British government ledgers originated from Singapore when it served as one of the centers of the India-China trade. The vast majority of these manuscripts suffer from ink-bleed that occurs when ink written on the document penetrates through the paper to become visible on the opposite side. Ink-bleed and ink-corrosion of old documents are serious problems affecting archives worldwide.

The severity and characteristics of ink-bleed, also referred to as bleed-through, is related to a variety of factors including the ink's chemical makeup, the paper's physical and chemical construction, the amount of ink applied and the paper's thickness (both spatially varying), the document's age, and the amount of humidity in the environment housing the documents. Figure 1-(a) shows examples of ink-bleed exhibiting various levels of severity and intensity characteristics from four different documents.

The current approach used by the archives to reduce ink-bleed is to place the document in a chemical wash that indiscriminately removes ink from the document. This physical restoration takes an entire month to correct a single bound volume, requiring all pages to be unbound, washed and dried, and then rebound. Further drawbacks include risk of damage to the source material due to handling, potential removal of lightly written foreground ink, and environmental hazards from harsh solvents. Not surprisingly, a digital solution that operates on images of the documents is desired as an alternative.

The motivation of our work is to provide a practical framework to reduce or remove ink-bleed in digital images. Figure 1-(b) shows examples where ink-bleed has been reduced by our system. The manuscripts targeted by our work exhibit a wide range of ink-bleed characteristics making it difficult to develop an automated approach that can work on all inputs. As a result, we have developed a user-assisted approach where small amounts of training-data are used per input. While engaging the user is a straight-forward idea, it is a significant departure from existing approaches in related literature that all strive for full automation. Our *expedite* versus *automate* strategy was not only acceptable to the archivists working with these materials, but when likelihood estimates from the training-data were combined with our Markov Random Field (MRF) formulation we produced outputs which were superior to existing approaches.

In this paper, the ink-bleed reduction problem is treated as classification where image pixels are labeled as either foreground, ink-bleed, or background. This pixel labeling is aided by a dual-layer MRF with smoothness cost designed to reduce noise while maintaining foreground strokes in regions where foreground and ink-bleed overlap. Our approach operates on a wide-range of ink-bleed and does not require assumptions about the ink-bleed intensity or extensive parameter tuning. All necessary components needed for this application are presented, including input pre-processing to align the front and back image, collection of the training-data, and the dual-layer MRF setup with associated data cost and intra- and inter-layer smoothness cost computations.

A shorter version of this work has appeared in [1] which

Y. Huang and D. Xu are with the School of Computer Engineering, Nanyang Technological University, Singapore, 639798.

M. S. Brown is with the School of Computing, National University of Singapore, Singapore, 117576.

focused on the dual layer MRF formulation together with a K-Nearest Neighbor (KNN) likelihood estimation. A related paper that discussed this framework in the context of the end users (i.e. archivist and archive patrons) appeared in [2]. The work in [2] focused on overall software including the user interface and post-processing tools useful for manual cleanup. In this journal version, we expand the work in [1] to provide more details on the choice of features used in classification as well as provide a comparison of KNN with support vector machine (SVM) classification to show that a slight gain in accuracy can be obtained with SVM. This journal version also provides more objective experiments via real and synthetic examples.

The remainder of this paper is organized as follows: section II discusses related work; section III provides an overview of our application framework; section IV-A and section IV-B discuss feature and classifier selection; section IV-C details the data-cost computations and dual-layered MRF formulation; section V presents results including comparisons against other approaches; section VI provides a discussion and conclusion.

II. Related Work

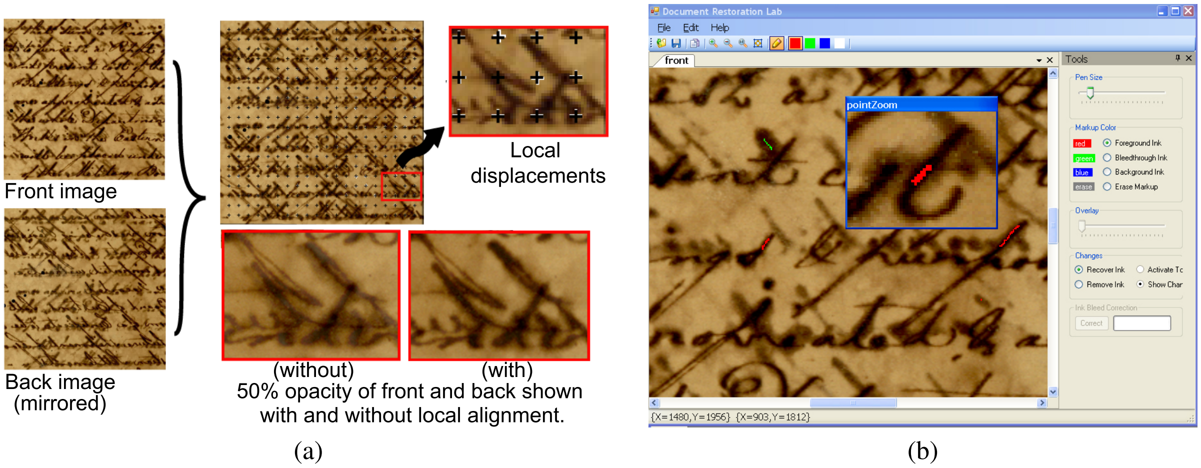
Some of the previous work addressing ink-bleed focuses on documents with relatively little ink-bleed interference. In such cases, the ink-bleed's intensity is clearly less than the foreground ink and can often be successfully reduced using either local or global thresholding techniques (e.g. [3], [4], [5]).

In this paper we address significantly more complex ink-bleed that is not suitable for standard thresholding. Similar types of complex ink-bleed are addressed in Drira et al [6], which separates foreground from ink-bleed. This approach works from a single RGB image whose color space is first reduced to its first two PCA components. Ink-bleed is then removed by iterative clustering and thresholding based on the PCA features where the first iteration decomposes the image into background and non-background regions, and subsequent iterations decompose the non-background region into foreground and ink-bleed. Work by Tonazzini et al [7] targeted complex ink-bleed using blind signal separation via Independent Component Analysis (ICA) which linearly decomposes an RGB image into three signals assumed to be foreground, background, and ink-bleed. This approach also targeted a single RGB image. Iterative thresholding and source separation approaches produce good results when the ink-bleed and foreground have clearly distinguishable graylevel intensities or RGB signatures. These techniques suffer when the ink-bleed and foreground have similar intensities as shown in some of the examples in Figure 1-(a). The iterative-thresholding technique make a further assumption that the ink-bleed grayscale intensity on one page is *always* lighter than the foreground on *the same page*, an often invalid assumption. Wolf [8] used an MRF framework with two hidden fields and one observation field to separate the foreground and ink-bleed from either RGB or grayscale images. This approach used Gaussian functions to model the color distributions of foreground, ink-bleed and background. However, we have found that a Guassian distribution assumption does not work for complex cases. The data term of their MRF was defined based on the clustering results from grayscale features, resulting in the dark ink-bleed pixels wrongly classified as background. We note that [6], [7], [8] attempt to reduce ink-bleed via only a single image which provides limited information.

One way to obtain more information is to use images from both the front and back side of a document. Sharma [9] demonstrated a successful two-image 'show through' reduction approach for use in xerox imaging. Show through, however, assumes global bleeding between the front and back images where ink-bleed typically varies spatially making ink-bleed more difficult to model. Tonazzini et al. [10] extended their blind-source separation approach [7] to operate on grayscale aligned front and back images and demonstrated its effectiveness for ink-bleed as well as show through. These approaches assumed that the ink-bleed was lighter in intensity than the foreground, limiting its range of input. In [11], Tonazzini et al. further extended their work by using MRF and ICA. Similar as in [10], [11] also assumed that the two signals are mixed by a linear operation, which is not satisfied for the old documents used in our work. In addition, their MRF was used to process a single image only.

Tan et al. [12] performed ink-bleed reduction in the wavelet-domain. These techniques first globally align the front and back images from which an initial classification of the foreground and ink-bleed strokes was made using the magnitude of the image difference. Iterative filtering of the wavelet coefficients is then used to dampen ink-bleed while sharpening foreground pixels. While this technique produces good results, six parameters must be tuned *per example*, including thresholds for the difference-image, the amount of dampening and sharpening of the wavelet coefficients, the number of wavelet scale-levels, and the number of iterations. A variation of [12] was presented by Wang et al. [13] with the assumptions that the foreground and ink-bleed stokes all slant at 45 and 135 degrees respectively, which is not valid for many images.

Our work is distinguished from previous approaches by its allowance for user-interaction and our dual layer MRF formulation. This gives us several key advantages. First, almost all previous approaches need to make prior assumptions about the ink-bleed characteristics, namely that ink-bleed intensity is lighter (i.e. higher intensity) than the foreground, or that the grayscale intensity/color values of ink-bleed, foreground, and background follow a parametric distribution, e.g. Gaussian. In our approach, the ink-bleed characteristics are embedded in the training data provided by the user and it does not require any particular parametric model. This allows our approach to work on a larger range of inputs without assumptions on intensity profiles. Second, our approach does not require parameter tuning making it significantly easier to use by the target users of archives and/or archive patrons who have little to no knowledge on image processing. Lastly, our MRF framework helps to reduce errors and maintain foreground stroke integrity by incorporating spatial information into solution.



Pen Size

Markup Color
red ◉ Foreground Ink
green ○ Bleedthrough Ink
blue ○ Background Ink
erase ○ Erase Markup

Overlay

Changes
◉ Recover Ink ○ Activate Tc
○ Remove Ink ● Show Char

Ink Bleed Correction
Correct

{X=1480,Y=1956} {X=903,Y=1812}

Fig. 2. Preprocessing steps of our framework: (a) alignment and (b) markup. (a) Local displacements computed between the front and back images. A zoomed inset shows front regions (black-crosses) matched to their corresponding locations in the back image (white-crosses). Overlapped image regions *with* and *without* local alignment are shown with 50% opacity. 'Ghosting', visible from misalignment, is removed with the local alignment procedure. (b) A simple painting interface is used for markup and ink-bleed removal. Features such as global and local zoom aid the markup.

III. Framework Overview

In this section, we discuss our overall application framework, including a brief description to our targeted users, input pre-processing involving front and back image alignment, and training-data collection via user markup. Details of feature selection, data cost initilization, and our dual layer MRF are given in section IV.

A. Application Usage

As mentioned in section I, this work is done in partnership with the National Archive of Singapore that houses hundreds of volumes of governmental ledgers, circa 1820-1860, suffering from ink-bleed. Many of these ledgers have already been imaged to grayscale microfilm while others are imaged upon request. Our application is intended as a post-processing tool to 'cleanup' the images to make the documents more legible. While most users are computer-literate, they have little to no background in computer vision or image processing. The usage model, and need for ink-bleed reduction, can then be considered an "as needed basis". While anecdotal evidence only, our discussion with other archives suggests that this usage pattern at the NAS is quite representative of other national archives. This makes the reliance on user-interaction in the solution significantly more acceptable than if batch processing was required. Figure 2-(b) shows the screen shot of the interface of the application. Our framework first aligns the front and back images of a document. After this, a paint-like interface is used to label regions containing foreground, ink-bleed, and background. From the labeled data, the final result is computed based on our dual-layer MRF framework.

B. Image Alignment Pre-Processing

Our framework starts with images ($\sim$1K$\times$2K) of the front and back side of a page. Images obtained from microfilm are grayscale, while flatbed scans of the original materials are RGB. The pages in these volumes are typically bound and as a result are not completely pressed flat when imaged. This non-planar imaging compounded with small 3D surface variations that are typical of older documents makes it impossible to align the front and back images with a single global transform. While 'flattening' techniques can be used to remove these 3D surface variations (e.g. [14], [15], [16], [17]), such approaches require additional 3D scanning equipment not available in mainstream imaging setups and cannot be used on existing microfilmed documents. As a result, a local alignment procedure in addition to global alignment is needed.

Our approach assumes that the input images are already coarsely aligned. If the alignment is significantly off (for each more than 20 pixels), our software tool allows the user to adjust the back's images position [2]. From input images, we first perform an initial global alignment to provide a good starting point for the local alignment. The global alignment procedure begins by mirroring the back image. The global displacement is computed by finding the maximum correlation score of the front and mirrored back image over a $[-20, 20]$ pixel range in the horizontal and vertical direction. For our examples, translational alignment suffices, however, a full affine alignment could be incorporated if rotation and minor scale changes are an issue.

Local displacements are computed by dividing the front image into local windows (60×60 for our examples). Correlation is performed between each front image window with its corresponding back location over a $[-10, 10]$ pixel range in the horizontal and vertical direction. The location of the maximum correlation score for each window is taken to be the local displacement. If the maximum score is below a minimum threshold, it is assumed that there is no local change. Using the local displacements, thin-plate-spline (TPS) interpolation [18] is used to warp the back page to align with the front. Figure 2-(a) shows this local-alignment procedure. The front and back image regions are shown overlapped with 50% opacity. Ghosting from misalignment is visible when local alignment is not performed; this is removed after TPS warping.

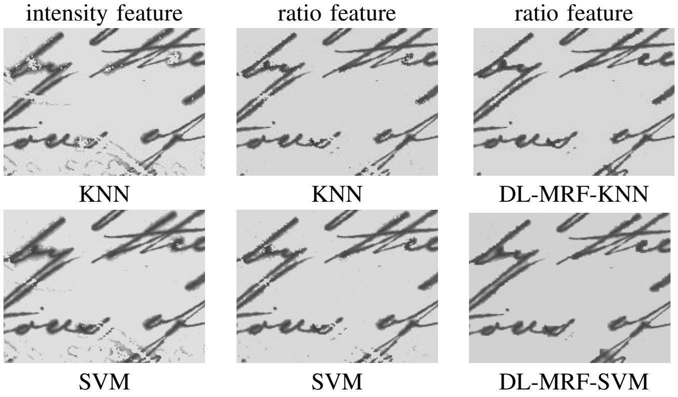


Fig. 3. Comparison of intensity feature and ratio feature. The three columns show a representative result using the grayscale intensity feature, ratio feature, and ratio feature combined with the dual layer MRF. The top row shows the results using KNN as the classifier, and the bottom row uses SVM. The ratio feature consistently produced better results in our experiments. The combination with the dual layer MRF further improved the results.

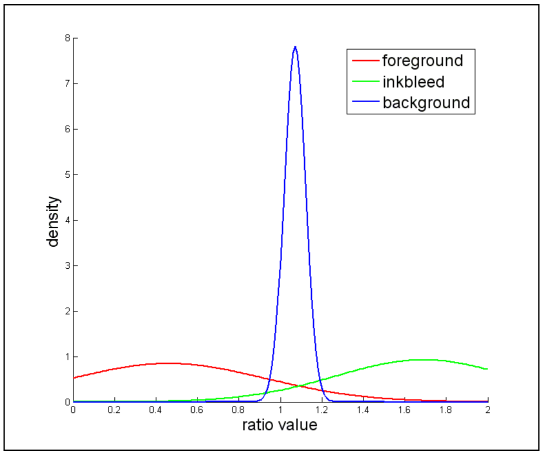


Fig. 4. Distributions of the training data in ratio feature space. For better illustration, we only show the distribution between 0 and 2 given the majority of training samples are within this range. Three Gaussian functions are used to model the distributions of the training data from three classes respectively for visualization(Red: foreground; green: ink-bleed; blue: background.)

This alignment approach works well when the ink-bleed interference is sufficiently strong. For inputs where ink-bleed interference is light, this approach can produce bad results. Currently, our framework has no mechanism to alert the user when the alignment has failed, however, for such cases, existing approaches that operate from a single image are sufficient to reduce the ink-bleed.

C. Training Data Collection

Our approach requires user assistance in labeling foreground, ink-bleed, and the document's background. A simple painting interface is used to draw color-coded strokes on the front and back images. Features such as local region zooming are provided to support the markup. Figure 2-(b) shows the markup of the front image of a document pair. Note that similar markup will be performed on the back image. Our interface also has a split-screen mode to allow the user to see the cursor on the front with its corresponding position on the opposite image.

IV. Classification

After image alignment and user-markup, the next goal of our framework is to label each pixel in the image as one of three classes: *Foreground*, *Ink-bleed* and *Background*, which are denoted as $\{\mathcal{F}, \mathcal{I}, \mathcal{B}\}$ respectively. Our framework comprises two stages. First, classification techniques such as K-Nearest Neighbor (KNN) or Support Vector Machine (SVM) are used for likelihood estimation. Second, we use a dual layer MRF formulation to label all the pixels in which the data-cost energy is defined as the likelihood from the KNN or SVM classifiers. We will refer to our dual layer MRF framework with SVM and KNN classifiers as DL-MRF-SVM and DL-MRF-KNN respectively. The details about classification methods and dual layer MRF are discussed in this section. We begin the section by first discussing the feature used by our classifiers.

A. Feature Selection

Choosing a good feature is crucial in classification. Our input are either RGB images (document scans) or grayscale images (microfilm). We empirically observed that using the full RGB color provided no advantage over using grayscale values alone and therefore we work only with the gray channel even for RGB images, i.e. we convert RGB images to grayscale images before further process. Ink-bleed arises due to interference from the ink on the opposite side of the document at the same position. Considering that we have aligned the pixels from the front and back images, one choice is to use a two dimensional intensity feature $\{C_p, C_{p'}\}$, where C_p and $C_{p'}$ are the grayscale intensity values of two corresponding pixels p and p' from both sides of an image. We have also found another good choice is to use the ratio of the two grayscale intensity values at the same position (referred to as ratio feature here), i.e., $\rho_p = \frac{C_p}{C_{p'}}$.

Figure 3 shows a representative result that compares grayscale intensity feature and ratio feature when used by both KNN and SVM classifier. While we only show a single image here, this trend was exhibited in all of our input data.

To give some insight into why the ratio feature works well, we show Figure. 4 that plots the distribution of ratio feature based on the labeled training data from one image. For better illustration, we only plot the distribution in the range [0,2] given that the majority of data falls within this range. We also use three individual Gaussian functions to model the distribution of the training data from three classes respectively (for visualization purposes only). While the samples from the three classes share some common regions, it is clear that they are mainly distributed at three different areas in the ratio space, *i.e.*, foreground data is mainly between zero and one, ink-bleed data is mainly larger than one, and background data is mainly around one.

We have tried numerous other features, including full RGB and ratio of RGB values (for scanned documents), combinations of grayscale intensity and ratio, difference between front and back, and image gradient information about each pixel. In the end, ratio of the intensity consistently produced the best result.

The ratio feature by itself, however, is not completely sufficient to produce an acceptable result. From the KNN and SVM results shown in Figure 3, there are remaining errors. For example some parts of the foreground strokes are misclassified as background or ink-bleed, and some of the non-foreground pixels are misclassified as foreground. This is because the classification methods treat each pixel independently. We will introduce a dual layer MRF framework in Section IV-C to enhance the spatial smoothness in which the data-cost energy is defined as the likelihood from KNN or SVM classifiers. The results from DL-MRF-SVM and DL-MRF-KNN are further improved as shown in Figure. 3, as well as our other experiments.

B. Classifiers

In our approach, the first step is to use the classification methods to generate the data-cost energy for our MRF. We consider two well known classifiers KNN and SVM and compare their effectiveness.

1) K-Nearest Neighbor: KNN classifies each feature vector according to the class labels of its k nearest neighbors from training data in feature space. The error rate of the nearest neighbor classifier is bounded at most twice as many errors as the optimal Bayes rule classifier [19], under the assumption that infinite training data are available. However, in our application, the total number of training samples from the user annotation is limited. We have directly applied KNN for classification based on the user labeled training data and observed that the classification results were not satisfactory. To overcome this problem, we first use KNN to label the entire image with the ratio feature, where K is empirically set as the square root of the size of the user-labeled data. For each class, we then add the 10% most confident pixels into the training set. This training-data enlargement procedure is also shown in Figure 5.

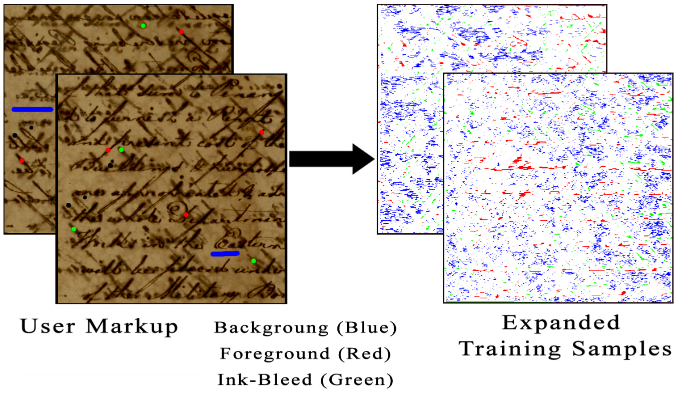


Fig. 5. The small amount of user-supplied training-data can be further expanded using highly confident pixels labeled using an initial KNN classifier. Original markup has been thickened for better visualization.

Based on the enlarged training set, we perform KNN to classify all the pixels again. To speedup the classification, we used K-means to cluster the training-data, with cluster centers of each class represented as $\{\rho_i^{\mathcal{F}}\}|_{i=1}^{L}$, $\{\rho_j^{\mathcal{I}}\}|_{j=1}^{M}$ and $\{\rho_k^{\mathcal{B}}\}|_{k=1}^{N}$. While choosing the optimal number of cluster centers is an open problem, we empirically set $L = M = N$ as 10% of the size of the smallest training-set. For each pixel p, we compute the Euclidean distances between ρ_p and all the $L + M + N$ cluster centers and then select the top-K closest centers where K is set as $\sqrt{L + M + N}$. The top-K centers are denoted as$\{\rho_m\}|_{m=1}^{K}$ and are further divided into three index sets $\pi^{\mathcal{F}}$,$\pi^{\mathcal{I}}$ and $\pi^{\mathcal{B}}$ according to their labels. The distance between ρ_p and the m-th cluster center ρ_m is computed by $d_{pm} = ||\rho_p - \rho_m||$. We also denote $d_p^2 = \frac{\sum_m d_{pm}^2}{K}$ as the mean squared distance to the top-K centers. The likelihood of pixel p to each class is defined as:

$$\begin{aligned} S_{\mathcal{F}} &= \sum_{m \in \pi^{\mathcal{F}}} \exp(-d_{pm}^2/d_p^2) \\ S_{\mathcal{I}} &= \sum_{m \in \pi^{\mathcal{I}}} \exp(-d_{pm}^2/d_p^2) \\ S_{\mathcal{B}} &= \sum_{m \in \pi^{\mathcal{B}}} \exp(-d_{pm}^2/d_p^2). \end{aligned} \tag{1}$$

2) Support Vector Machine: Another classifier applied in our application is SVM, which has been successfully used in many applications, such as object recognition, content-based image retrieval and so on. SVM aims to achieve a minimal structural risk by simultaneously minimizing the average risk on the training set (*i.e.*, Empirical Risk) and controlling the model complexity. For a binary classification task, the decision function for a test sample p has the following form:

$$g(p) = \sum_i \alpha_i y_i K(\rho_i, \rho_p) - b, \tag{2}$$

where ρ_i and ρ_p are the ratio features of training pixel i and test pixel p, $K(\rho_i, \rho_p)$ is the value of a kernel function for the two ratio values, y_i is the class label of pixel i ($+1$ or -1), α_i is the learned weight of the training sample i, and b is the threshold parameter. The training samples with weight $\alpha_i > 0$ are called support vectors. The support vectors and their corresponding weights are learned using the standard quadratic programming optimization process or other variations.

In our implementation using *libsvm* [20], we use the Radial Basis Kernel, $K(x, y) = \exp(-\gamma\|x - y\|^2), \gamma > 0$, where the optimal parameter γ is determined by five-fold cross-validation. To cope with three-class classification task in this work, we adopt the one-against-rest method by training three SVMs, in which the training data from one class and the rest two classes are treated as positive and negative samples respectively. Each pixel p is classified by the three SVMs using Eq (2), resulting in three decision values: $v_p^{\mathcal{F}}$, $v_p^{\mathcal{I}}$ and $v_p^{\mathcal{B}}$. Based on the decision values, we can calculate the similarities (or probabilities) $S_{\mathcal{F}}$, $S_{\mathcal{I}}$ and $S_{\mathcal{B}}$ by sigmoid function:

$$\begin{aligned} S_{\mathcal{F}} &= \frac{1}{1 + \exp(-v_p^{\mathcal{F}})} \\ S_{\mathcal{I}} &= \frac{1}{1 + \exp(-v_p^{\mathcal{I}})} \\ S_{\mathcal{B}} &= \frac{1}{1 + \exp(-v_p^{\mathcal{B}})}, \end{aligned} \tag{3}$$

3) KNN versus SVM: For our application KNN and SVM produced similar results, with SVM slightly outperforming KNN. There are pros and cons for using each. As a lazy learning algorithm, KNN has the advantage of simple implementation without any training process. In addition, KNN tends to the optimal Bayes rule classifier as the size of the training sample increases to infinity. The main disadvantage of KNN is that it can not be generalized too much beyond the labeled data set, i.e., KNN can not achieve satisfactory performance when the total number of training data is limited. To address this problem, we therefore used KNN twice to first expand the training data and then compute the likelihood based on the enlarged training data.

As the state-of-the-art classification method, SVM has been shown (theoretically and empirically) to have excellent generalization capabilities by minimizing the structural risk. But a training process is required in SVM for model learning, in which time-consuming cross-validation is utilized to determine the optimal parameters (e.g., γ in Radial Basis Kernel and regularization parameter in SVM).

However, both SVM and KNN are used to classify individual pixels with no spatial mechanism to enforce spatial coherency. This results in small noises throughout the result. In addition, it results in foreground strokes that often appear broken, especially in regions where foreground and ink-bleed overlay. To deal with these issues we introduce the following MRF formulations.

4) Discussion about the Training Data: As shown in Fig. 2, the user input in our system is very simple. For most of the whole page images with resolution of ∼1K×2K, 10-15 strokes of markup as shown in Figure 8 are enough to achieve good results by employing either KNN or SVM. With less strokes (*e.g.*, five strokes) provided by the users, the results are slightly worse for some complex examples, but still acceptable. For more challenging cases such as the spatial varying images, the users need to specify the diverse background pixels from different areas (*e.g.*, one or two strokes from each area) to achieve better results.

C. Dual Layer Markov Random Field

After classification using only feature values, all pixels can be formulated as a Markov Random Field (MRF) framework for labeling using their intrinsic spatial correspondence. To label each pixel into $\{\mathcal{F},\mathcal{I},\mathcal{B}\}$, the task is formulated as a discrete labeling MRF where each pixel, p is assigned a label l_p, where $l_p \in \{\mathcal{F},\mathcal{I},\mathcal{B}\}$ (see [21] for details of MRF formulations). As shown in [21], the optimal label assignment can be found by minimizing the following energy terms:

$$E = E_d + \lambda E_s, \tag{4}$$

where E_d represents the data-cost energy associated with the likelihoods of assigning an l_p to each pixel and E_s is a smoothness energy based on the MRF's prior cost for assigning neighboring pixels different label values. The scalar weight λ is to balance the two terms, which is set as one in our work. While this energy function is standard for all MRFs, the associated likelihood (data cost) and the prior (smoothness cost) are unique for each problem. Details of E_d are given in section IV-C1.

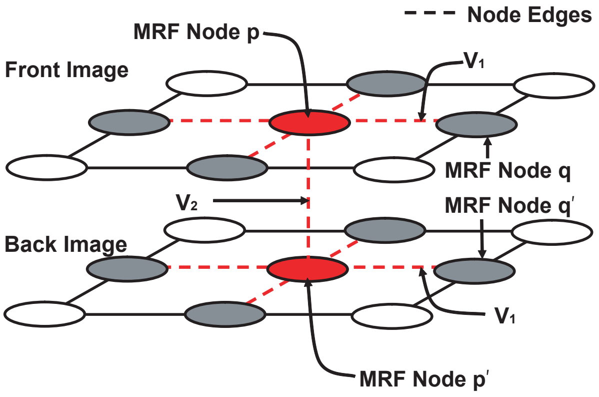


Fig. 6. Our MRF network with associated nodes and edges.

Our smoothness term E_s is composed of intra-layer edge costs, $V_1(l_p, l_q)$, that computes the cost of assigning neighboring pixels the labels l_p and l_q and inter-layer edge costs (between layer), $V_2(l_p, l_{p'})$, that computes the cost of assigning a label combination to pixel p and its corresponding pixel p' on the opposite layer. Intra-layer edges are set for both the front and back image, thus we also have edges $V_1(l_{p'}, l_{q'})$ as shown in Figure 6. Intra-layer edge costs are designed to encourage consistent labels for neighboring pixels based on feature likelihoods. The inter-layer edge costs are designed to avoid invalid label configurations and aid in resolving regions with overlapping ink. This dual layer combination proves significantly more effective at maintaining foreground strokes compared with using the intra-layer alone. Details to E_s are given in section IV-C2.

1) Data Cost Energy E_d: The data-cost E_d is defined for both the front and back image. Only the front is described here for example. The data-cost term, E_d, for each label is defined as:

$$\begin{aligned} E_d(l_p = \mathcal{F}) &= \frac{S_{\mathcal{I}} + S_{\mathcal{B}}}{2 \times (S_{\mathcal{F}} + S_{\mathcal{I}} + S_{\mathcal{B}})} \\ E_d(l_p = \mathcal{I}) &= \frac{S_{\mathcal{F}} + S_{\mathcal{B}}}{2 \times (S_{\mathcal{F}} + S_{\mathcal{I}} + S_{\mathcal{B}})} \\ E_d(l_p = \mathcal{B}) &= \frac{S_{\mathcal{F}} + S_{\mathcal{I}}}{2 \times (S_{\mathcal{F}} + S_{\mathcal{I}} + S_{\mathcal{B}})}. \end{aligned} \tag{5}$$

in which the likelihood values of each pixel are calculated by KNN or SVM classifier discussed in SectionIV-B. Eq(5) results in E_d ranging between zero and one, and $E_d(l_p = \mathcal{F}) + E_d(l_p = \mathcal{I}) + E_d(l_p = \mathcal{B}) = 1$.

2) Smoothness term E_s: As previously stated, the prior term E_s is computed as edge costs within a layer and between layers, namely:

$$E_s = \sum_{p,q \in \mathcal{N}} V_1(l_p, l_q) + \sum_{p,p' \in \mathcal{M}} V_2(l_p, l_{p'}), \tag{6}$$

where $p, q \in \mathcal{N}$ are the within layer edges and $p, p' \in \mathcal{M}$ are the between layer edges. These two terms are weighted equally.

Intra-Layer Edge Costs: Intra-layer costs are based on the intensity difference or ratio difference between two intra-layer neighbors p and q. We define $d^{\rho}_{pq} = ||\rho_p - \rho_q||$ and $d^{c}_{pq} = ||C_p - C_q||$ as the distance between p and q in terms of ratio and intensity feature respectively. We normalize d^{ρ}_{pq} and d^{c}_{pq} to range between zero and one. To impose smoothness constraints in the intra-layer while preserving the edges between different classes, the intra-layer cost is expressed as:

$$V_1(l_p, l_q) = \frac{1}{1 + (\xi_{pq})^2}, \tag{7}$$

where ξ_{pq} is defined in the following table:

l_p	l_q		
	Foreground	Ink-Bleed	Background
Foreground	∞	d^{ρ}_{pq}	d^{c}_{pq}
Ink-Bleed	d^{ρ}_{pq}	∞	d^{ρ}_{pq}
Background	d^{c}_{pq}	d^{ρ}_{pq}	∞

It is worthwhile to note that we use d^{c}_{pq} to define the intra-layer cost in *Foreground-Background* configuration because we observe that the intensity from the foreground and background pixels differ more than that of the ratio feature. In other configurations, we use the default ratio feature. Moreover, if the neighbors have the same label, we use zero cost to enforce the smoothness constraint (the three ∞ in the diagonal cells result in zero costs).

Inter-Layer Edge Costs: Inter-layer costs, $V_2(l_p, l_{p'})$, are defined as:

l_p	$l_{p'}$		
	Foreground	Ink-Bleed	Background
Foreground	0	0	0
Ink-Bleed	0	∞	∞
Background	0	∞	2ω

In the above table, we have a conditional constraint for the *Background-Background* configuration. We set ω as a constant 1 if ($C_p < C^1_{avg}$ and $C_{p'} < C^2_{avg}$), 0 otherwise, where C^1_{avg} and C^2_{avg} are the average intensities of the foreground pixels labeled by KNN or SVM classifier before MRF procedure in the front and back images respectively. The background pixels are usually the brightest pixels in the whole document, thus we assume that both front and back pixel that have lower intensity (i.e. darker) are not likely to be a *Background-Background* configuration and a small weight of 2 is used as penalization. We use three *infinity* penalties in the above table for invalid configurations. For example, if one pixel is labeled as ink-bleed, the corresponding pixel in the opposite image can only be foreground. All other configurations are possible and are therefore assigned a zero cost.

The only two parameters in our MRF framework are λ and ω, which are used to balance two energies and enforce small penalty for the unlikely *Background-Background* configuration. In this work, we empirically fix λ and ω as 1, and our experiments show that this setting works well on all the documents with wide spectrum of ink-bleed. It may be possible to compute the optimal parameters based on the input images' intensity profiles, however, this task is currently left for future work.

3) Minimizing the Objective Function Energy: Several state-of-the-art energy minimization approaches (see [22]) can be used to minimize our global objective function stated in Eq (4). We use the α-expansion move of graph cuts approach to optimize our energy function (see [23], [24], [25]). The Middlebury's MRF code provided by [22] is modified to incorporate our dual layer configuration. In all of our experiments, we find that satisfactory results can be obtained with five iterations.

V. Results

Experiments are reported using images from the National Archives of Singapore. For the real images we first show subjective results on representative ink-bleed examples. We compare our approach with the representative algorithms in different approaches, as well as our own approach using only a single layer of the MRF. Since ground truth is not available for the real images, an objective evaluation is performed that counts of the number of *Missed*, *Interfered*, and *Added* words. In order to provide a ground truth comparison, a synthetic testcase is used that computes accuracy at pixel level. These experiments are described in the following.

Note that user-markup varies per example, but it generally consists of no more than $5 - 15$ strokes or points drawn on both the front and back images.

A. Real examples

1) Subjective Results: Several results on images from the NAS are shown here. Results shown in this paper have no further post-processing of any kind. Pixels labeled as foreground are shown with the original image intensity, all other pixels are set to the mean color of the background training samples.

Figure 7 shows sub-regions from two examples that represent a reasonably diverse range of ink-bleed. Shown are the original front and back input, results from the single-image adaptive thresholding approach by Drira et al [6], the two-image based wavelet domain approach by Tan et al [12], the KNN approach with single and dual layer MRF (SL-MRF-KNN and DL-MRF-KNN), the SVM approach with single and dual layer MRF (SL-MRF-SVM and DL-MRF-SVM). Figure 8 shows a full-page example with comparisons of selected regions shown at the bottom.

For all examples, the dual layer approaches DL-MRF-KNN and DL-MRF-SVM provide much better performance, when compared with the prior work from single-image adaptive thresholding and wavelet approach. For the results from the prior work, we observe that some parts of the foreground strokes are missing as well as obvious noise, which are the non-foreground pixels misclassified as foreground by the prior approaches. The wavelet approach [12] produces good results in some cases, but requires six-parameters to be tuned per example in order to obtain the best results.

The results from DL-MRF-KNN and DL-MRF-SVM show that almost all the foreground strokes are correctly preserved. In addition, there are less noisy pixels remaining in the image. We also observe that the DL-MRF-KNN and DL-MRF-SVM outperform SL-MRF-KNN, and SL-MRF-SVM respectively, which demonstrates that a single-layer MRF is not sufficient

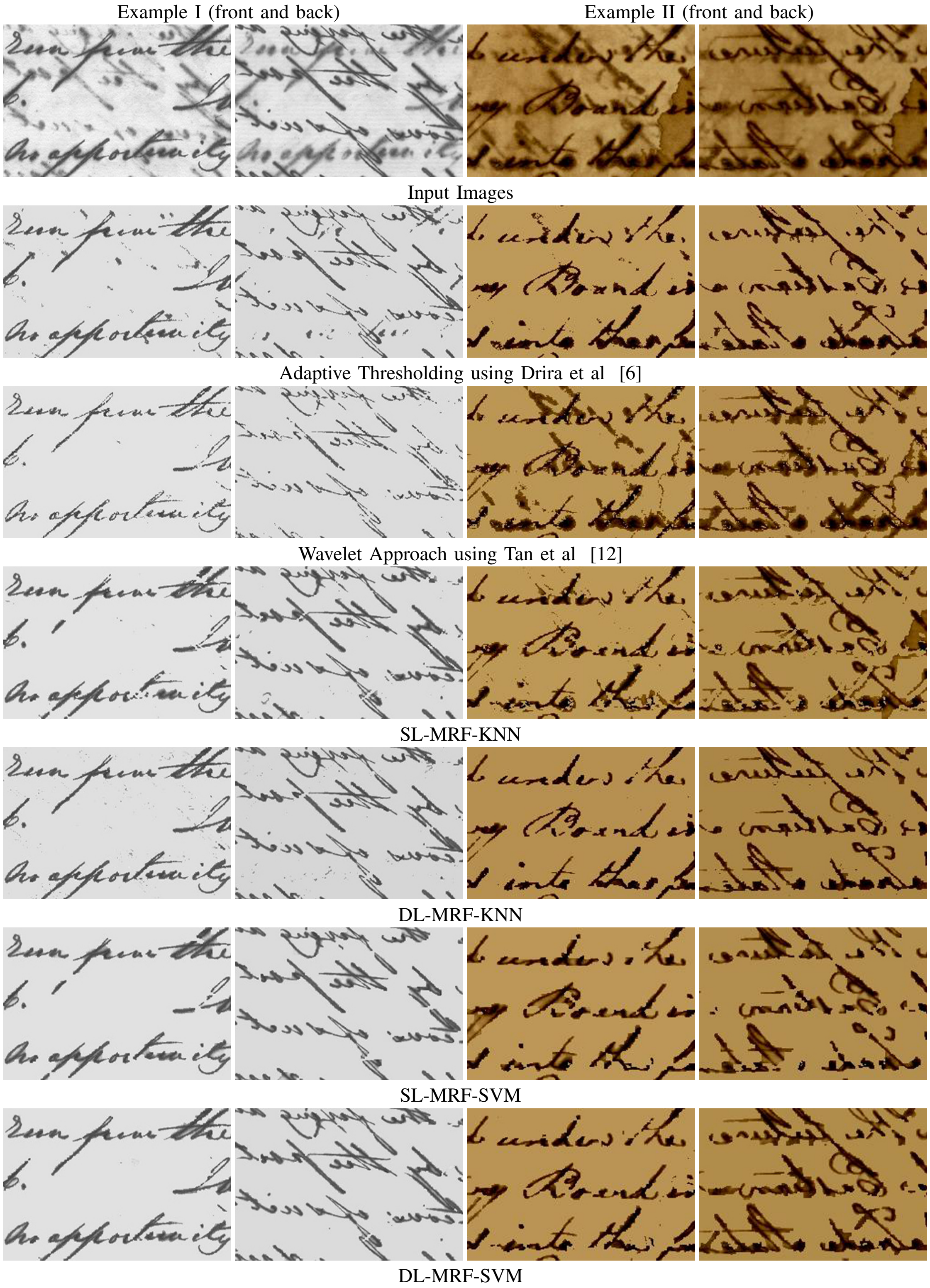

Example I (front and back)
Example II (front and back)
Input Images
Adaptive Thresholding using Drira et al [6]
Wavelet Approach using Tan et al [12]
SL-MRF-KNN
DL-MRF-KNN
SL-MRF-SVM
DL-MRF-SVM

Fig. 7. Two representative example pairs showing both the front and back image. The back image is mirrored for clarity. Results are shown for the adaptive thresholding by Drira et al [6], the Wavelet approach by Tan et al [12], KNN with single layer MRF, KNN with our dual layer MRF, SVM with single layer MRF, and SVM with dual layer MRF.

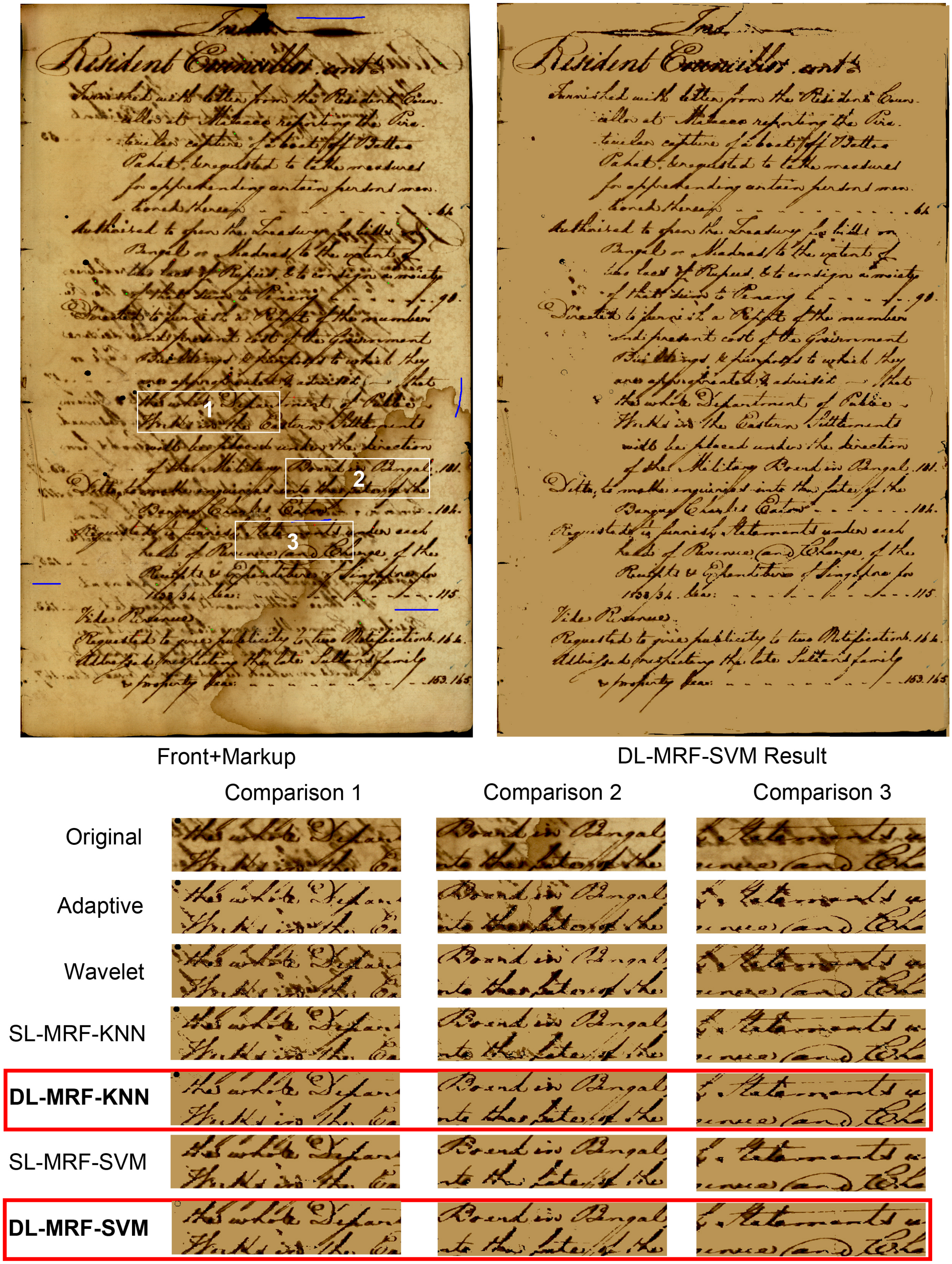


Fig. 8. A full page example (RGB scan) with only the front is shown (top) with the complete user markup. Comparisons with other techniques are shown for three selected regions. Our dual layer MRF approaches produce the best results.

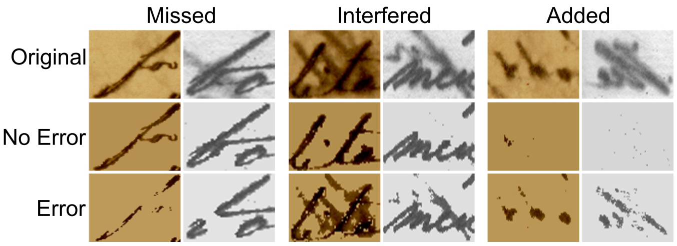


Fig. 9. Examples of *Missed*, *Interfered*, and *Added* errors in objective evaluation using real images. Shown are the input images (top row), examples that would be categorized as having *no error* (middle row), and examples of *errors* (bottom row).

for ink-bleed reduction. Finally, we observe that DL-MRF-SVM slightly outperform DL-MRF-KNN, in particular the foreground strokes in DL-MRF-SVM are thicker resulting in better preserved foreground strokes.

2) Quantitative Results: As a quantitative evaluation on the real images, we compute the number of words having errors observed in the output images.

At a document level two types of errors can occur: 1) foreground words not detected correctly; 2) ink-bleed or background words detected as foreground. Sets of words exhibiting these errors are denoted as W^F and W^B respectively, with W denoting the total number of foreground words in a document. W^F can be divided into two groups: *Missed* and *Interfered*. *Missed* words refer to foreground words that are not classified as foreground or have large portions missing. *Interfered* refers to words where large amounts of the ink-bleed are labeled incorrectly with the foreground strokes, typically occurring when ink-bleed overlaps with foreground strokes. W^B is composed of *Added* words and refers to pixel regions (most often ink-bleed) that are incorrectly labeled as foreground. Figure 9 shows examples of these three types of errors. Following the definitions from Wang et al [12], recall and precision are defined as:

$$\text{recall} = \frac{W-W^F}{W} \qquad \text{precision} = \frac{W-W^F}{W-W^{Missed}+W^B} \tag{8}$$

where $W - W^F$ is the correctly recognized (or understood) words, and $W - W^{Missed} + W^B$ is the total detected words by the system both correctly and incorrectly [1].

Ten results containing more than 1200 words in total were selected to represent a wide spectrum of ink-bleed. Performing word-by-word comparison among four different approaches is a non-trivial task, and great care was taken to be fair when counting correct and erroneous words. The results are presented in Figure 10, together with precision and recall for five front back image pairs (ten images in total), against other techniques. From a user's point of view, *Missed* errors are the most serious as missing a word can potentially change the document's meaning. For some examples, (i.e. trial 2, 8, 9) the number of missed words are high, however, most of these are in fact 'partially' missed words, where the word may still be recognizable as shown in Figure 9.

Given recall and precision, F measure can be used to evaluate the total performance. According to [26], it is defined as:

$$F_\beta = \frac{(1+\beta^2)\times(precision \times recall)}{\beta^2 \times precision + recall}, \tag{9}$$

where β is the weight to balance the precision and recall. Considering that it is more intolerable for the users to have a word missing than to have ink-bleed still remaining, we give recall more weight than precision. We report the results from the classical F_2 measure in Figure 10, in which β is set as 2, making recall weighted twice as much as precision.

Again, we observe that dual layer MRF based approaches (DL-MRF-KNN and DL-MRF-SVM) achieve the best results in all categories. In addition, DL-MRF-SVM slightly outperforms DL-MRF-KNN.

B. Synthetic Experiment

As there is no ground truth for our archival images, we designed a pair of synthetic images as shown in Figure 11. Each word has spatially-varying intensity values ranging from $60 - 120$, which represents difficult cases from our archival images. The background is left as white. We observe that the documents provided by the NAS contain very little noise and the serious problem is with the ink-bleed. We therefore only consider ink-bleed in the synthetic images. In our synthetic images, we exaggerate the variation in intensity within foreground and ink-bleed by generating images with ink-bleed darker than the foreground in some parts of the image and lighter than the foreground in other parts of the image.

For every pixel p, its possible ink-bleed intensity I_p^b is generated by its opposite pixel p''s original intensity $I_{p'}^o$. As there is no established model of bleeding process, we simply assume the severity of I_p^b is proportional to the darkness of p', which means that the darker the pixel is, the more ink it bleeds to the opposite side. As darker pixels have smaller intensity values, we use $(255 - I_{p'}^o)$ to represent the darkness of the pixel. Ink-bleed intensity appearing at pixel p is defined as:

$$I_p^b = 255 - (255 - I_{p'}^o)\exp\left(-\left(\frac{I_{p'}^o}{t}\right)^n\right), \tag{10}$$

where t is an intensity threshold that pixels with intensity below can cause bleed. This is done because light foreground text does not cause ink-bleed. The term $\exp\left(-(\frac{I_p}{t})^n\right)$ is a value in $[0, 1]$ which means the portion of the original darkness appear in the ink-bleed. n is a parameter which controls the ink-bleed variation speed. If n is large, when the strokes become lighter, the ink-bleed is lightened very fast and vice versa. The intensity of a pixel p in the synthetic image is defined as:

$$I_p = \min(I_p^o, I_p^b), \tag{11}$$

which means if the ink-bleed caused by the back side is lighter than pixel at front side, the front side will not be affected.

In our experiments, we empirically set $t = 150$ and $n = 6$ and generate the synthetic images shown in Figure 11. Although this model may not perfectly describe the real ink-bleed

[1] In the standard definition of precision, the denominator is $W - W^F + W^B = W - W^{Missed} - W^{Interfered} + W^B$. While interfered words can not be correctly understood by the annotator, they are still detected as foreground words by the system. Therefore, we do not subtract $W^{Interfered}$ in the denominator of Eq. (8).

Algorithm		Trial 1	Trial 2	Trial 3	Trail 4	Trail 5	Trail 6	Trail 7	Trail 8	Trail 9	Trail 10	Averages
	# of words	171	141	121	115	110	109	124	112	115	126	
Adaptive	Missed	44	66	4	12	7	7	8	40	9	23	Recall 69.16%
	Interfered	18	18	7	12	17	9	13	17	40	22	Precision 86.67%
	Added	4	17	9	19	8	0	0	21	31	13	F_2 Measure 72.07%
Wavelet	Missed	4	9	3	19	12	20	7	28	44	22	Recall 74.03%
	Interfered	64	68	2	7	0	3	0	6	8	10	Precision 94.16%
	Added	9	21	0	2	0	0	0	5	6	9	F_2 Measure 77.34%
SL-MRF-KNN	Missed	7	54	9	13	22	33	7	37	46	22	Recall 76.23%
	Interfered	17	7	2	3	2	0	6	1	3	1	Precision 97.72%
	Added	3	4	1	4	0	0	0	0	4	6	F_2 Measure 79.74%
DL-MRF-KNN	**Missed**	**4**	**33**	**2**	**7**	**8**	**9**	**3**	**17**	**17**	**7**	**Recall 87.31%**
	Interfered	**10**	**8**	**5**	**2**	**0**	**0**	**4**	**4**	**9**	**10**	**Precision 98.03%**
	Added	**3**	**5**	**2**	**1**	**0**	**0**	**0**	**0**	**6**	**5**	**F_2 Measure 89.26%**
SL-MRF-SVM	Missed	5	66	7	19	25	34	13	48	34	40	Recall 75.11%
	Interfered	7	0	1	1	0	0	0	0	2	1	Precision 99.21%
	Added	2	0	0	0	0	0	1	0	1	4	F_2 Measure 78.95%
DL-MRF-SVM	**Missed**	**2**	**24**	**3**	**4**	**9**	**10**	**0**	**14**	**14**	**9**	**Recall 90.36%**
	Interfered	**5**	**5**	**1**	**4**	**0**	**0**	**5**	**2**	**4**	**6**	**Precision 98.57%**
	Added	**1**	**3**	**0**	**1**	**0**	**0**	**6**	**0**	**1**	**5**	**F_2 Measure 91.89%**

Fig. 10. The number of *Missed*, *Interfered*, and *Added* words resulting from the following approaches: single-image adaptive thresholding [6], the two-image wavelet-based approach [12], the SL-MRF-KNN, DL-MRF-KNN, SL-MRF-SVM, and DL-MRF-SVM.(See text for details and examples of *Missed*, *Interfered*, and *Added* words).

process, we believe it generates a reasonable (and challenging) synthetic example for testing.

1) Subjective Results: The results in Figure 11 show our dual layer MRF approaches outperform other techniques. Portions of the foreground strokes are lost using Drira et al [6] while they are broken using Tan et al [12]. On the other hand, our dual layer MRF works well using either KNN or SVM.

2) Quantitative Results: For objective evaluation, we define precision and recall similarly to those in Section V-A2 on pixel basis. Then there are only two error scenarios: (1) foreground pixels mislabeled as ink-bleed or background and (2) ink-bleed or background pixels that are mislabeled as foreground, which are denoted as P^F and P^B respectively. Based on the ground truth, the correctly detected foreground pixels are denoted as $P - P^F$. We also denote the total number of foreground pixels in ground truth as P. Precision and recall at pixel level are defined as:

$$\text{recall} = \frac{P - P^F}{P} \qquad \text{precision} = \frac{P - P^F}{P - P^F + P^B}, \tag{12}$$

Figure 12 shows the quantitative evaluations on the results of synthetic images. Both KNN with dual layer MRF and SVM with dual layer MRF outperform other techniques in terms of both precision and recall. Note that in the pixel-level evaluation, $P - P^F$ is used to represent the correctly detected foreground pixels, which can be easily determined by using the ground truth.

C. Result Discussion

As with any system that relies on user input, the user needs to provide enough training data that represent the variations within each class. For most cases in our experiments, 10-15 dots and stokes are enough for an entire page as shown in Figure 8. Note the non-geometric damage such as the water stain in Figure 8 can also be eliminated if the user marks some pixels in the water stain regions as background. If there is spatial varying foreground or ink-bleed in the image, it is better for the user to give markups covering the whole range of every class. For example, if there are both light and dark foreground strokes on the same page, marking one stroke for each type as foreground would be enough. Even if there is spatial variation in the image, 4-5 markup strokes for each class are enough to cover all the variation.

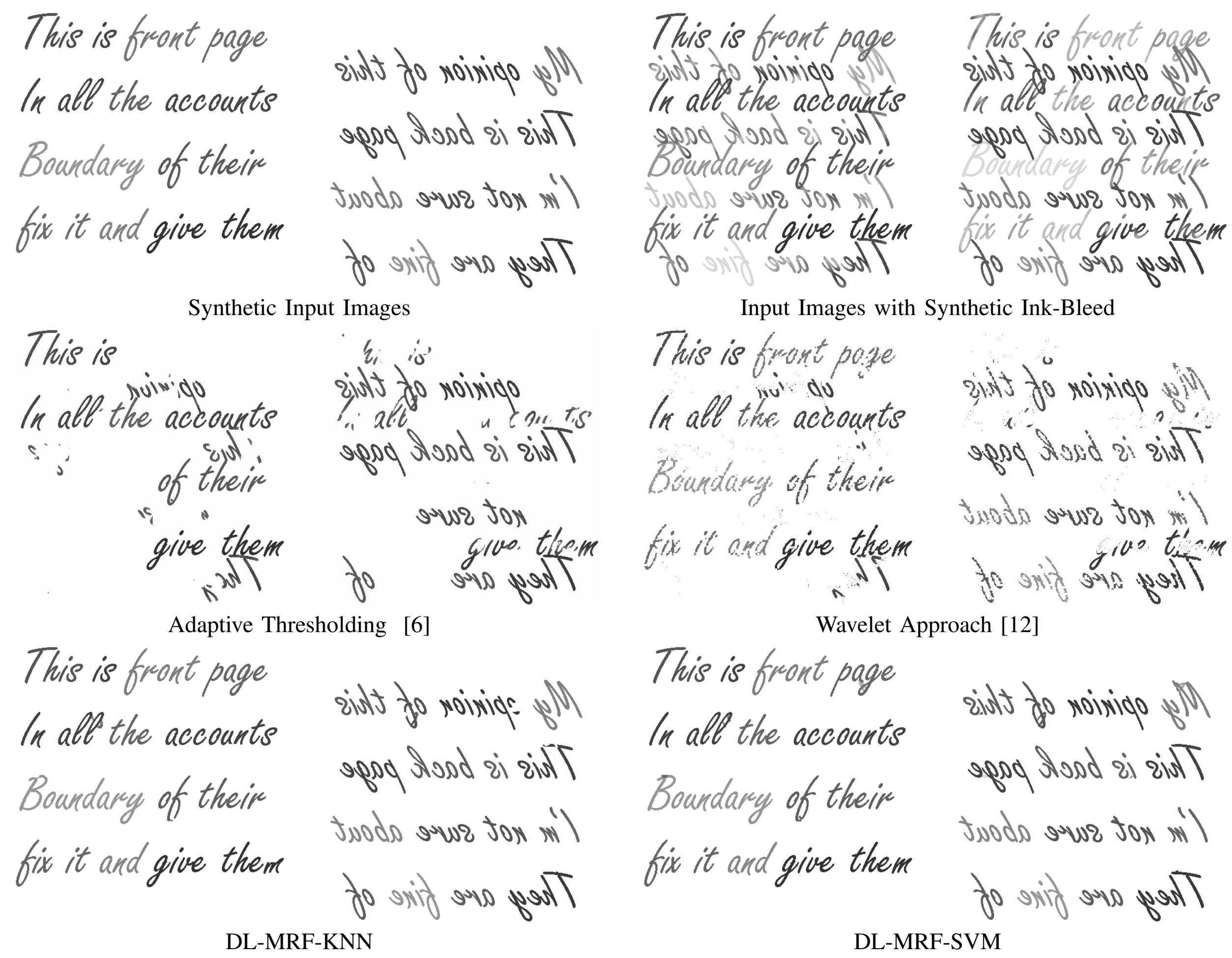
Considering the processing time is image dependent, we tested 10 pairs of images with size $1K \times 1K$ on a 2.67GHz multi-core CPU and computed the average time to process an image pair. For adaptive thresholding [6], it takes around 4.8 seconds to process a pair. The wavelet based approach [12] takes around 29.9 seconds. For our approaches, on average it takes 167.9 seconds for SL-MRF-KNN and 187.1 seconds for DL-MRF-KNN, 147.2 seconds for SL-MRF-SVM and 165.0 seconds for DL-MRF-SVM. Although adaptive thresholding is very fast, the results are not satisfactory. While our method seems slower than wavelet based approach [12], we note that in our experience the wavelet based approach [12] requires several attempts at tuning the six parameters. Thus, we believe our processing times are acceptable in this application given the alternatives.

VI. Discussion and Conclusion

Our results demonstrate that our DL-MRF-SVM and DL-MRF-KNN can generate results superior to previous approaches. As discussed in Section II one key reason for this improved performance is our allowance of user-interaction in the application framework which avoids the need for assumptions about ink-bleed characteristics. Furthermore, our MRF formulation helps to enforce spatial coherency and disallow invalid configurations in the final results.

As to the exact amount of markup needed, this of course differs from input to input. In addition, different markup can produce similar results. As with other supervised learning approaches, our approach does assume that the user labeled data is correct. Validation techniques can be used to help throw out outliers in the event of erroneous user markup especially along the boundary of foreground and ink-bleed strokes.

In conclusion, we have presented a novel user-assisted framework for reducing ink-bleed. Our framework falls within



Synthetic Input Images

Input Images with Synthetic Ink-Bleed

Adaptive Thresholding [6]

Wavelet Approach [12]

DL-MRF-KNN

DL-MRF-SVM

Fig. 11. A synthetic example. Synthetically generated images and simulated ink-bleed (top row), results of ink-bleed reduction using [6], [12] (middle row), and our DL-MRF-KNN and DL-MRF-SVM (bottom row).

	Adaptive		Wavelet		**DL-MRF-KNN**		**DL-MRF-SVM**	
	Front	Back	Front	Back	**Front**	**Back**	**Front**	**Back**
Precision	73.44	74.29	88.96	87.64	**99.51**	**99.60**	**99.15**	**98.52**
Recall	57.22	73.12	80.97	81.51	**99.13**	**99.00**	**99.63**	**99.80**
Average Precision	73.87		88.30		**99.56**		**98.84**	
Average Recall	65.17		81.24		**99.07**		**99.72**	
F_2 Measure	66.67		82.56		**99.17**		**99.54**	

Fig. 12. Quantitative evaluations based on pixel-level ground truth using synthetic images. Recall, precision, and F_2 measure are reported.

the recent trend of so called "interactive computer vision", where problems that are too hard to automate, or ill-posed, are aided with the user's help. By combining KNN and SVM likelihood computations together with our dual layer MRF we are able to produce results superior to previous work. We believe this framework provides a practical approach to ink-bleed removal that can target a wide range of examples and is suitable for use in a real-world setting.

VII. Acknowledgements

This work was supported by the Singapore National Research Foundation Interactive Digital Media Research and Development Program, under research Grant NRF2008IDM-IDM004-018 and MOE AcRF Tier-1 Grant RG63/07.

References

[1] Y. Huang, M. Brown, and D. Xu, "A framework for reducing ink-bleed in old documents," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008.

[2] Y. Huang and M. Brown, "User-assisted ink-bleed correction for handwritten documents," in *IEEE/ACM Joint Conference on Digital Libraries*, 2008.

[3] J. Bescos, "Image processing algorithms for readability enhancement of old manuscripts," *Electronic Imaging*, vol. 1, pp. 392–397, 1989.

[4] F. C. M. et al., "Toward on-line, worldwide access to vatican library materials," *IBM Journal of Research and Development*, vol. 40, no. 2, pp. 139–162, March 1996.

[5] Z. Shi and V. Govindaraju, “Historical document image enhancement using background light intensity normalization,” in *International Conference on Pattern Recognition*, 2004.

[6] F. Drira, F. L. Bourgeois, and H. Emptoz, “Restoring ink bleed-through degraded document images using a recursive unsupervised classification technique.” in *Document Analysis Systems (DAS’06)*, 2006, pp. 38–49.

[7] A. Tonazzini, L. Bedini, and E. Salerno, “Independent component analysis for document restortion,” *International Journal on Document Analysis and Recognition*, vol. 7, pp. 17–27, 2004.

[8] C. Wolf, “Document ink bleed-through removal with two hidden markov random fields and a single observation field,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 99, no. 1, 2009.

[9] G. Sharma, “Show-through cancellation in scans of duplex printed documents,” *IEEE Transactions. on Image Processing*, vol. 10, no. 5, pp. 736–754, 2001.

[10] A. Tonazzini, E. Salerno, and L. Bedini, “Fast correction of bleed-through distortion in grayscale documents by a blind source separation technique,” *International Journal on Document Analysis and Recognition*, vol. 10, pp. 17–27, 2007.

[11] A. Tonazzini, L. Bedini, and E. Salerno, “A markov model for blind image separation by a mean-field em algorithm,” *IEEE Transactions on Image Processing*, vol. 15, February 2006.

[12] C. L. Tan, R. Cao, and P. Shen, “Restoration of archival documents using a wavelet technique,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 10, pp. 1399–1404, Oct 2002.

[13] Q. Wang, T. Xia, L. Li, and C. Tan, “Document image enhancement using directional wavelet,” in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2003.

[14] M. Brown and W. B. Seales, “Image restoration of arbitrarily warped document,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 10, pp. 1295–1306, 2004.

[15] M. Brown and Y. Tsoi, “Geometric and shading correction of printed materials,” *IEEE Transactions on Image Processing*, vol. 15, no. 6, June 2006.

[16] M. Pilu, “Undoing paper curl distortion using applicable surfaces,” in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2001.

[17] C. L. Tan, Z. Li, Z. Zhang, and T. Xia, “Restoring warped document images through 3d shape modeling.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 2, pp. 195–208, 2006.

[18] F. Bookstein, “Principal warps: Thin-plate splines and the decomposition of deformations,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 6, pp. 567–585, June 1989.

[19] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience Publication, 2000.

[20] C. Chang and C. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[21] S. Li, *Markov Random Field Modeling in Image Analysis (2nd Edition)*. Springer-Verlag, 2001.

[22] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother., “Comparative study of energy minimization methods for markov random fields,” in *European Conference on Computer Vision*, 2006.

[23] Y. Boykov and V. Kolmogorov, “An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004.

[24] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, November 2001.

[25] V. Kolmogorov and R. Zabih, “What energy functions can be minimized via graph cuts?” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147–159, February 2004.

[26] C. J. van Rijsbergen, *Information Retrieval*. Butterworth, 1979.