Handwritten Carbon Form Preprocessing Based on Markov Random Field

Huaigu Cao and Venu Govindaraju
Center for Unified Biometrics and Sensors (CUBS)
Dept. of Computer Science and Engineering
University at Buffalo, Amherst, NY
{hcao3, govind}@buffalo.edu

Abstract

This paper proposes a statistical approach to degraded handwritten form image preprocessing including binarization and form line removal. The degraded image is modeled by a Markov Random Field (MRF) where the prior is learnt from a training set of high quality binarized images, and the probabilistic density is learnt on-the-fly from the gray-level histogram of input image. We also modified the MRF model to implement form line removal. Test results of our approach show excellent performance on the data set of handwritten carbon form images.

1. Introduction

Our work is motivated by preprocessing badly degraded handwritten document images, such as carbon forms, for recognition and retrieval. Carbon form recognition is commonly considered as a very hard, or even impossible problem. This is largely due to the extremely low image quality. Usually the quality of a document image is affected by varying illumination and noise such as Gaussian noise, artifacts, smearing, and so on. By assuming that the background changes slowly, the problem of varying illumination has been solved by some adaptive binarization algorithms such as [14], Niblack [13] and Sauvola [16]. Although noise can be depressed by smoothing, the resulting blurring will also affect the OCR rate. Approaches based on heuristics, to name a few, Kamel/Zhao [9], Yang/Yan [19], and Milewski [12], solve the problem to some extent by heuristic search of stroke locations.The Kamel/Zhao algorithm is a local algorithm which finds stroke locations and then removes the noise in the non-stroke area using an interpolation and thresholding step. A parameter of stroke width is needed. The Yang/Yan algorithm is a variant of the method by Kamel/Zhao which is meant to handle varying intensity, illumination, and smearing. The Milewski algorithm is also a heuristic based method. It detects strokes from local statistics in different directions. In recent years, inspired by the success of Markov Random Field (MRF) in the area of image restoration [2], [3], [4], some attempts were made to apply MRF to the preprocessing of text region of degraded images [5], [6], [18]. The advantage of the MRF model over heuristics is that it can describe the probabilistic dependency of neighboring pixels or image patches, i.e., the prior probability, and learn it from training data.

In order to use MRF, one needs to pick forms of prior and observation models. Usually this is done in ad hoc way. The forms of MRF's taken by all the existing approaches dealing with textual image are not very appropriate for handwritten documents. The MRF based approach proposed by Wolf *et al*. [18] defined the prior model on a 4×4 clique and is appropriate for textual images in low resolution video. However, for 300 dpi high resolution handwritten document images, it is not feasible to learn the prior probability or energy potentials if we simply define a much larger neighborhood. Gupta *et al*. [5], [6] studied restoration and binarization of blurred images of license plate digits. They adopted the factorized form of MRF, *i.e*., the product of compatibility functions [2], [3], [4]. They defined compatibility functions as mixtures of multivariate normal distributions calculated over samples of their training set, and incorporated recognition into the MRF to reduce the number of samples involved in the calculation of compatibility functions. However this scheme can hardly be applied to unconstrained handwriting image because of the larger number of classes and the low performance of existing handwriting recognition algorithm. In this paper we propose an MRF based approach to degraded handwritten document image preprocessing. We use the MRF with the same topology as adopted in [2], [3]. Different from existing MRF based algorithms for textual image preprocessing [5], [6], [18], our algorithm uses a collection of standard patches, or representatives to represent each patch of the binarized image from the test set. These representatives are obtained by clustering all patches of binarized images in the training set. Use of representatives reduces the domain of the prior model to a very

1-4244-1180-7/07/$25.00 ©2007 IEEE

limited size. We are not going to use our model to solve the problem of image restoration from linear or non-linear degradation so do not need an image/scene pair for learning the observation model. By assuming additive noise the observation model is learnt on-the-fly from the local histogram of the test image which ensures that our algorithm gets performance close to well-known adaptive thresholding algorithms when omitting the prior model, and the result gets improved later with the spatial constraints added by the prior model.

In addition to binarization, we also apply our algorithm to removal of form lines. This process is also referred to as image inpainting and is done by inferring missing portion of images from spatial constraints, for which the MRF is very suitable. For a heuristic approach to line removal please refer to [21]. For some works related to inpainting of natural scene images please refer to [1], [20]. We applied the same document image specified MRF model as that we use for binarization, with only a few modifications of the way calculating probabilistic density of the observation model, to paint in the region of unwanted form lines, title lines, and machine-printed text.

2. Markov Random Field model for handwritting images

As shown in figure 1, we use an MRF model with the same topology as [3]. In our model, a binarized image x is divided into non-overlapping square patches $x_1, x_2, ..., x_N$, and the input image, or the observation y is also divided into patches $y_1, y_2, ..., y_N$ so that x_i corresponds to y_i for any $1 \leq i \leq N$. Each binarized patch is statistically dependent on its four neighboring patches in both horizontal and vertical directions. Each observed patch is statistically dependent on its corresponding binarized patch. The model can be considered as a graph where each node represents a patch of the input or binarized image and each edge represents the dependency of two patches. Later we will see the advantage of a patch based structure in that relatively larger area of the local image are statistically dependent and it is also possible to find some way to compress dimensionality of patches.

Given the posterior probability $P(x|y) = \frac{P(x,y)}{P(y)}$, since $P(y)$ is a constant over x, the objective would be to find $\underset{x}{\text{argmax}} P(x, y)$. Solving an MRF involves two phases: a learning phase for learning the parameters of the network connections, and an inference phase for estimating the binarized image of the input image.

2.1. Inference of MRF

Inference of the MRF model can be achieved by first writing the MRF in the factorized form of the product of

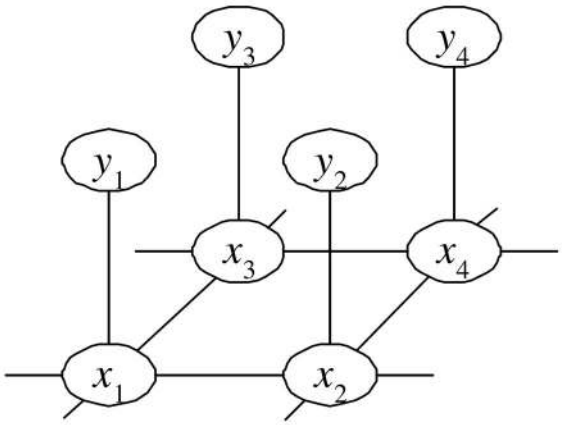


Figure 1. The topology of the Markov network. Each node x_i in the field is connected to its four neighbors. Each observation node y_i is connected to node x_i. An edge indicates the statistical dependency of two nodes.

compatibility functions of neighboring nodes (patches) (see [3] Equation (1)), and then running a local message passing algorithm known as belief propagation (BP) [15]. For an acyclic graph the BP algorithm will get the optimal MAP or MMSE estimate of x. In the case of graphs with cycles, BP was shown empirically to have excellent performance. An explanation of why BP provides good estimates was given in [17]. We nevertheless find that a simple form of compatibility function defined over the distance between the adjacent or overlapping portion of two neighboring patches as in [3] may not be proper for the case of binarized image because the distance can only take very small number of values. We finally choose another form of inference proposed in [3]. According to the factorization that resembles the factorization of the compatibility function form, that is, the factorization of the joint probability $P(x_1, x2, ..., x_N, y_1, y_1, ..., y_N)$ into conditional probabilities of neighboring nodes, Freeman *et al*. [3] suggested the following MAP estimation and message passing rules:

$$M_j^k = \max_{x_k} P(x_k|x_j) P(y_k|x_k) \prod_{l \neq j} \tilde{M}_k^l, \tag{1}$$

$$\hat{x}_{j\,\text{MAP}} = \underset{x_j}{\text{argmax}} P(x_j) P(y_j|x_j) \prod_k M_j^k, \tag{2}$$

where k runs over all four neighboring nodes of node j in the binarized image, M_j^k is the message from node k to node j, and $\tilde{M}_k^l$ is M_k^l from the previous iteration. The initial $\tilde{M}_j^k$'s are set to column vectors of 1's.

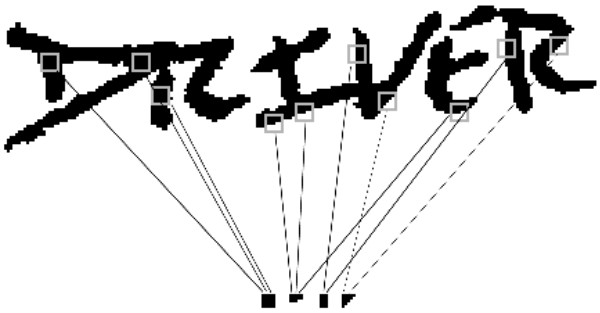


Figure 2. Shared patches in binary document image.

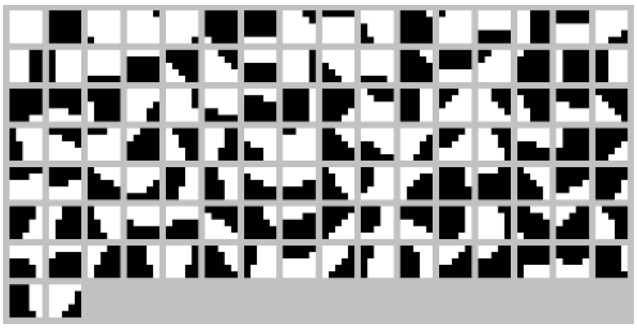


Figure 3. 114 representatives of shared patches obtained from clustering.

2.2. Learning the prior model

To use Equations (1), (2), the probabilities $P(x_k|x_j)$ and $P(x_k|x_j)$ have to be estimated. We can first estimate $P(x_j)$ and $P(x_j, x_k)$ in two directions, then use equation $P(x_k|x_j)$=$\frac{P(x_j,x_k)}{P(x_j)}$ to get the estimation of $P(x_k|x_j)$.

Note that if the size of the patch is $B \times B$, the variable x_j in $P(x_j)$ can take 2^{B^2} different values, and the pair (x_j, x_k) in $P(x_j, x_k)$ can take 2^{2B^2} different values. This makes the search for maximum in Equation (1) impossible. A plausible solution could be to search only the values of patches within a training set [3]. We instead take another strategy that depends less on samples in training set and has better reduction ratio. Our method is inspired by the idea that images of similar objects can be represented by very small number of shared patches in spatial domain. A recent work [8] explored the possibility of representing an image by a smaller miniature composed of shared patches. Owing to the fact that handwriting images, especially binary images, with fixed pen-width under the same resolution can be decomposed into patches that appear frequently (see figure 2), we apply similar strategy to the binarized patches. In particular, the B^2-dimensional binarized patches are represented by limited number of standard patches, or representatives. In this sense, this process can be considered as a vector quantization.

The representatives are learnt through clustering of all patches involved in training and are defined as centers of clusters $\{\mu_1, \mu_2, ..., \mu_M\}$. We use the standard c-means clustering algorithm with modifications that the distance from a sample vector to a mean vector is represented by the Hamming distance between them, and that each component of a mean vector calculated after every iteration, which is a real number, is rounded into $\{0, 1\}$ so the mean patch is still binary.

We need to determine the size of patches B and the number of clusters. As we know, a larger patch size provides stronger local dependency, whereas it is hard to represent too large patches because of variety of writing styles of different writers. Given a training set of B by B binary patches $\{p_i\}$, run c-mean clustering with parameter M = 1024, and remove any duplicated cluster and any cluster containing less than η samples, where η is a parameter of the algorithm and takes the value of 1000 in our experiment. Then we get a set of cluster mean vectors μ=$\{\mu_1, \mu_2, ..., \mu_M\}$. The error of the representation is measured by equation

$$\epsilon_\mu = \frac{\sum_i \frac{\min_j \mathrm{hd}(p_i, \mu_j)}{B^2}}{|\{p_i\}|} \tag{3}$$

where $\mathrm{hd}(p_i, \mu_j)$ is the Hamming distance between p_i and μ_j, and $|\cdot|$ denotes the number of elements in a set. In our experiment, we collected about 2 million patch images from three high quality handwriting images from different writers for learning patch representatives. We tried different values of B ranging between 5 and 8 which coincide the range of stroke width in 300dpi handwriting image, and chose the largest value of B while maintaining the representation error ϵ_μ less than 0.01. Finally we got the patch size $B = 5$, representation error $\epsilon_\mu = 0.0079$, and 114 representatives (figure 3). The use of representatives yields a compression ratio of about 4 in terms of dimensionality which is equivalent to $2^{\frac{3}{4}B^2}$ times of compression of the capacity of the vector space.

Note: here we only learn the prior model, so patches involved in learning are from images of handwritings written on clear paper and are of almost the same text size but of better quality than our test images of carbon copies. Some samples for training are shown in figure 4.

The prior probability $P(x_j)$ ($x_j \in \mu = \{\mu_k\}$) is estimated by the following equations

$$\hat{P}(x_j) = \begin{cases} \dfrac{1}{\#\{\mu_m | \operatorname{hd}(p_i, \mu_m) = \min\limits_l \operatorname{hd}(p_i, \mu_l)\}}, \\ \dfrac{1}{\#\{p_i\}}, \text{if } \operatorname{hd}(p_i, x_j) = \min\limits_l \operatorname{hd}(p_i, \mu_l)\} \\ 0, \text{otherwise} \end{cases} \tag{4}$$

where $\#\{\cdot\}$ denotes the number of elements in set $\{\cdot\}$. In equation (4), in the case that there are multiple nearest cluster centers from which the distances to sample p_i are identical, the factor $|\{\mu_m | \operatorname{hd}(p_i, \mu_m) = \min\limits_l \operatorname{hd}(p_i, \mu_l)\}|$ in the denominator is introduced to distribute the contributions of sample p_i evenly to those cluster centers.

$P(x_j, x_k)$ are estimated in horizontal and vertical directions, respectively. $P(x_j, x_k)$ $(x_j, x_k \in \mu = \{\mu_k\})$ in horizontal direction is estimated by the following equation

$$\begin{aligned} &\hat{P}(x_j, x_k) \\ &= \begin{cases} \dfrac{1}{\#\left\{ (\mu_{m1}, \mu_{m2}) \middle| \begin{array}{l} \operatorname{hd}(p_{i1}, \mu_{m1}) = \min\limits_l \operatorname{hd}(p_{i1}, \mu_l), \\ \operatorname{hd}(p_{i2}, \mu_{m2}) = \min\limits_l \operatorname{hd}(p_{i2}, \mu_l) \end{array} \right\}}, \\ \dfrac{1}{\#\{p_{i1}, p_{i2}\}}, \text{if } \operatorname{hd}(p_{i1}, x_j) = \min\limits_l \operatorname{hd}(p_{i1}, \mu_l), \text{and} \\ \operatorname{hd}(p_{i2}, x_k) = \min\limits_l \operatorname{hd}(p_{i2}, \mu_l)\} \\ 0, \text{otherwise} \end{cases} \end{aligned} \tag{5}$$

where p_{i1} is the left neighbor of p_{i2}.

$P(x_j, x_k)$ $(x_j, x_k \in \mu = \{\mu_k\})$ in horizontal direction is estimated by similar equations except that p_{i1} is the top neighbor of p_{i2}.

2.3. Learning the observation model

Unlike applications such as super resolution, image deblurring, or general image restoration problem, we can assume that an ordinary document image is free of linear degradation (out-of-focus blur, motion blur, *etc.*) and only consider the degradation due to additive noise. Thus, for the observation model of a single pixel one may consider using the histogram based model in [18]. For a patch based observation model, we need to map the single-pixel version to the patch vector space. Given the distribution of the lightness of foreground (strokes) $p_f(y)$ and the distribution of the lightness of background $p_b(y)$, the conditional p.d.f $P(y_j|x_j)$ is calculated by

$$P(y_j|x_j) = \prod_{x_{j,j'}=0} p_b(y_{j,j'}) \prod_{x_{j,j'}=1} p_f(y_{j,j'}) \tag{6}$$

where $x_{j,j'}$ runs over all elements in patch vector x_j. $x_{j,j'}$ equal to 0 or 1 means pixel $x_{j,j'}$ belongs to the background

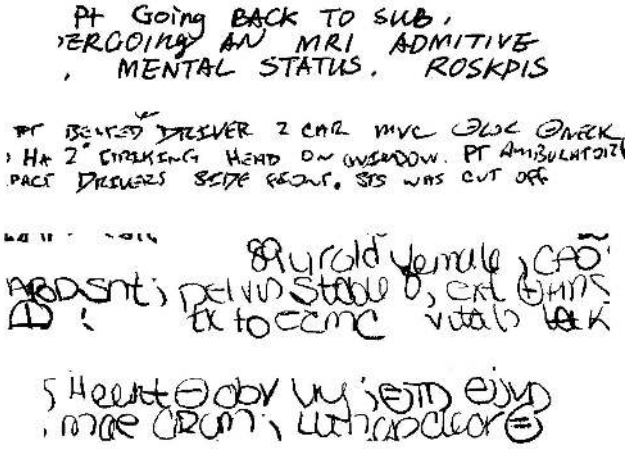


Figure 4. Binarized images from three writers for learning the prior model.

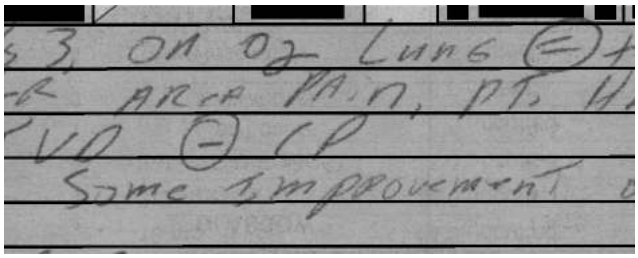


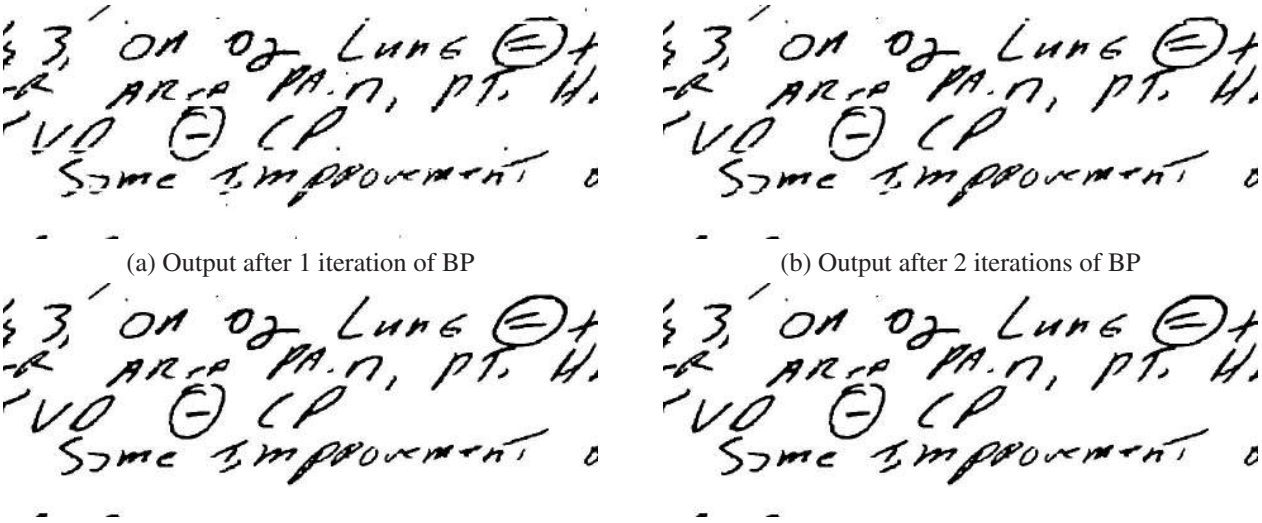
Figure 5. A sample patch cropped from a carbon image in our test set.

or foreground, respectively. And $y_{j,j'}$ is the observation of $x_{j,j'}$. Assuming that $p_f(y)$ and $p_b(y)$ are two normal distributions, we estimate $p_f(y)$ and $p_b(y)$ as follows. First determine a threshold T by an adaptive thresholding such as Niblack algorithm. Then use all pixels with gray-level $\leq T$ to estimate the mean and variance of $p_f(y)$. Use other pixels to estimate the mean and variance of $p_b(y)$. Thus, while ignoring the prior model but considering solely the observation model, the MRF model is reduced to an adaptive binarization algorithm.

We need only make a minor modification to equation (6) for line removal. We introduce a boolean matrix $\operatorname{hole}(j, j')$ to indicate if the pixel $y_{j,j'}$ is a hole for painting in. Then

$$P(y_j|x_j) = \prod_{\substack{x_{j,j'}=0, \\ \operatorname{hole}(j,j')=false}} p_b(y_{j,j'}) \times \prod_{\substack{x_{j,j'}=1, \\ \operatorname{hole}(j,j')=false}} p_f(y_{j,j'}) \tag{7}$$

The probability $P(y_j|x_j)$ in Equation (7) is 1 if $m(j, j')$ is always $true$ for any j'.



(a) Output after 1 iteration of BP

(b) Output after 2 iterations of BP

(c) Output after 4 iterations of BP

(d) Output after 16 iterations of BP

Figure 6. The binarization and line removal result of the sample shown in figure 5.

3. Experimental results and discussions

The proposed preprocessing algorithm is tested on a database of scanned carbon copy form images of NYS Prehospital Care Reports (PCR). The carbon copy images in the corpus are rather noisy and have faint carbon strokes. Besides, form cells often intersect texts. So very low word recognition rate (below 20%) was reported in [12]. This database contains very important medicare information of patients. The further objective would be to index and search the database.

First we applied our algorithm to the input image shown in figure 5. This input image is cropped from a PCR form. The original image is smoothed by a Gaussian blurring of radius = 0.5. Then lines and unwanted machine-printed region are identified and marked in black. Also notice that the our test images and images for training the prior model are from different writers. It is clear that the writing style in figure 5 is not like any of the styles in 4. The results after iterations 1, 2, 4, and 16 of belief propagation run on figure 5 are shown in figure 6. After the first iteration, the message has not yet been passed between neighbors. We can see that the edges of strokes are crooked due to noisy background and error of the vector quantization discussed in section 2.2, and all of the lines are dropped. After 2 iterations, text edges are smoothed but most lines are not well fixed. At the end of the 4th iteration, nearly all the strokes broken be lines are perfectly fixed, but we can still find a few glitches inside the circled equal sign. After 16 iterations the glitches are gone.

In the above test we adopted one more strategy for speeding up. At the end of the first and second iterations, while estimating $\hat{x}_{j\,\mathrm{MAP}}$ using equation (2), prune any x_j such that

$$\frac{P(x_j)P(y_j|x_j)\prod_k M_j^k}{\sum_{x_j} P(x_j)P(y_j|x_j)\prod_k M_j^k} < 0.1\% \qquad (8)$$

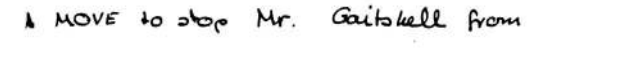
from the searching space of x_j. The size of test sample in figure 5 is 940 $\times$ 370. It took around 15 seconds to run the algorithm on a PC with an Intel 2.8G Hz CPU. This is pretty slow compared with traditional binarization algorithms. However, considering the hardness of the application, this is still acceptable. On the same image, we tested a modified algorithm that does not fix lines but further prunes all x_j's containing black pixels if

$$\frac{P(\mu_1)P(y_j|\mu_1)\prod_k M_j^k}{\sum_{x_j} P(x_j)P(y_j|x_j)\prod_k M_j^k} > 0.2, \qquad (9)$$

where μ_1 is the vector form of 5 by 5 white patch. It took 6 seconds to run the modified algorithm and we got similar performance except that lines were not removed. However when we used equation (9) for line removal, it performed badly. Due to missing of observations, estimated posterior probability of the true match returned in initial iterations is seldom among the largest, thus pruning of the searching space becomes harder.

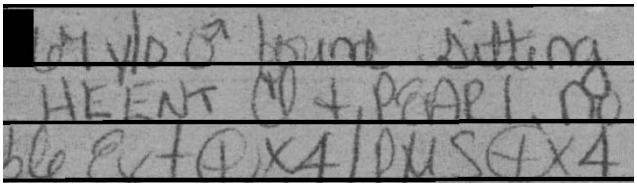
We also run the MRF binarization algorithm on some images from another well-known handwriting database IAM DB3.0 [11] (figure 7). Since the image quality in IAM DB is very good, our method, however, does not show much difference from other methods.

(a) Input image.

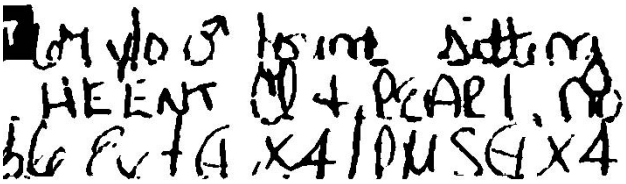


(b) Binarised image.

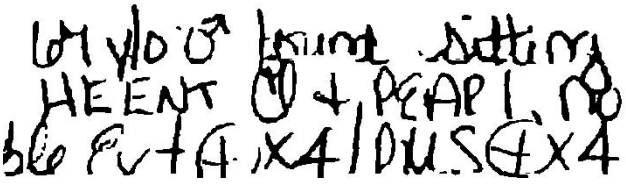
Figure 7. Binarization result of a sample from IAM database.



(a) Input image.



(b) Output of heuristic based algorithm.

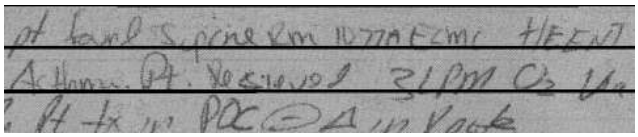


(c) Output of MRF based algorithm.

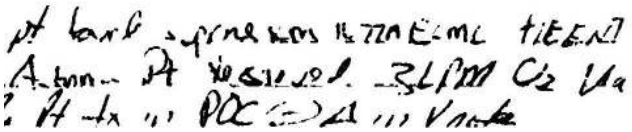
Figure 8. Comparison of the results of heuristic and MRF based algorithms (sample of fair quality).

The proposed approach is compared with the preprocessing algorithm of Milewski *et al.* [12]. Besides a heuristic based binarization, they also proposed a line removal method based on gray-scale interpolation from pixels neighboring to the line. They reported a performance of their binarization algorithm on the PCR form dataset which is better than classic methods such as Otsu, Niblack, Kamel/Zhao, and Yang/Yan algorithms in terms of word recognition rates.

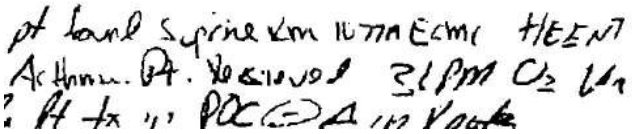
First we compare the output of two algorithms intuitively in figures 8 and 9. From figure 8 we can see that the output of proposed algorithm in (c) has less broken strokes than the output of heuristic algorithm in (b) even in outside of form lines, and fixes form lines more smoothly. An sample of really bad quality is tested and shown in figure 9. We can see that the MRF based algorithm is still able to produce a readable binary image, whereas in output of the heuristic



(a) Input image.



(b) Output of heuristic based algorithm.



(c) Output of MRF based algorithm.

Figure 9. Comparison of the results of heuristic and MRF based algorithms (sample of bad quality).

based algorithm, some words like "supine" and "HEENT" are hardly recognizable.

We also use OCR test to verify the effectiveness of our algorithm. We extracted two sets of image pairs of English words from the binarized images given by Milewski *et al.*'s and our algorithms. Set #1 contains 201 word image pairs that are not evidently affected by form lines, *i.e.*, no intersection. Set #2 contains 151 pairs that are affected by form lines. All the word images were recognized using the word recognition algorithm proposed in [10] with a lexicon of 4299 English words. We calculated top-n ($n \geq 1$)recognition rates instead of only the top-1 rate for comparison because top-n rates are of great significance to the problem of indexing text with very high error rate [7]. Moreover, recognition rates measured in terms of multiple candidates provides a strong proof of the effectiveness of the preprocessing. The resulting recognition rates are shown in table 1. Among the rates of set #1, although the top-1 rates of two methods are close, the top-2, 5, and 10 rates show that the proposed method has better performance. Rates of set #2 show distinct difference between two methods and indicate that the MRF based method provides more efficient line removal. The MRF based method also results in higher overall recognition rates.

4. Conclusions

In this paper we have presented a novel method for binarizing degraded carbon copy images of handwriting forms and removing form lines. Our method model the binarized objective image as a Markov Random Field. Different from related approaches, we reduce the large searching space of

Method		Heuristic	MRF
Set #1	Top 1 rate	14.4%	14.9%
	Top 2 rate	16.9%	21.4%
	Top 5 rate	22.8%	29.4%
	Top 10 rate	29.9%	38.3%
Set #2	Top 1 rate	21.9%	29.1%
	Top 2 rate	26.5%	35.1%
	Top 5 rate	37.7%	45.7%
	Top 10 rate	43.0%	53.0%
Overall	Top 1 rate	17.6%	21.0%
	Top 2 rate	21.0%	27.3%
	Top 5 rate	29.3%	36.4%
	Top 10 rate	35.5%	44.6%

Table 1. Comparison of word recognition rates of heuristic and MRF based approaches (set #1: sample word images not effected by forms lines; set #2: sample word images effected by forms lines; overall: set #1 + set #2).

the prior model to a class of 114 representatives, and learn the observation model directly from input image. Our work is the first attempt of applying stochastic method to the preprocessing of badly degraded carbon forms of handwritten data. The restriction of our model might be that it is essentially based on document image, but does not handle intense illumination variation, complicated background, and blurring that are common in low resolution video or pictures. However it is possible to generalize the model for more applications. Besides, there are some other issues concerning speeding-up the MRF, training multiple models to deal with different resolutions. We will investigate resolutions to these problems in our future work.

References

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. *Computer Graphics (SIGGRAPH 2000)*, pages 417–424, 2000. 2

[2] W. T. Freeman and E. C. Pasztor. Learning low-level vision. *Proc. of International Conference on Computer Vision*, pages 1182–1189, 1999. 1

[3] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *International Journal of Computer Vision*, 40(1):25–47, 2000. 1, 2, 3

[4] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984. 1

[5] M. D. Gupta, S. Rajaram, N. Petrovic, and T. S. Huang. Restoration and recognition in a loop. *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005. 1

[6] M. D. Gupta, S. Rajaram, N. Petrovic, and T. S. Huang. Models for patch based image restoration. *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, 2006. 1

[7] N. R. Howe, T. M. Rath, and R. Manmatha. Boosted decision trees for word recognition in handwritten document retrievals. In *Proceedings of the SIGIR*, pages 377–383, 2005. 6

[8] N. Jojic, B. J. Frey, and A. Kannan. Epitomic analysis of appearance and shape. *Proceedings of the Ninth IEEE International Conference on Computer Vision*, 2003. 3

[9] M. Kamel and A. Zhao. Extraction of binary characters/graphics images from grayscale document images. *CVGIP: Graphic Models Image Processing*, 55(3), 1993. 1

[10] G. Kim and V. Govindaraju. A lexicon driven approach to handwritten word recognition for real-time applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:366–379, April 1997. 6

[11] U. Marti and H. Bunke. The iam-database: an english sentence database for off-line handwriting recognition. *Int. Journal on Document Analysis and Recognition*, 5:39–46, 2006. 5

[12] R. Milewski and V. Govindaraju. Extraction of handwritten text from carbon copy medical form images. *Document Analysis Systems 2006*, pages 106–116, 2006. 1, 5, 6

[13] W. Niblack. An introduction to digital image processing. *Englewood Cliffs, N.J. Prentice Hall*, 1986. 1

[14] N. A. Otsu. A threshold selection method from gray-level histogram. *IEEE Transactions on System Man Cybernetics*, 9(1), 1979. 1

[15] J. Pearl. Probalistic reasoning in intelligent systems: networks of plausible inference. *Morgan Kaufmann Publishers Inc.*, 1988. 2

[16] J. Sauvola, T. Seppanen, S. Haapakoski, and M. Pietiktinen. Adaptive document binarization. *Proceedings of the 4th International Conference on Document Analysis and Recognition*, pages 147–152, 1997. 1

[17] Y. Weiss. Belief propagation and revision in networks with loops. *Technical Report 1616, MIT AI lab Cambridge MA 02139*, 1997. 2

[18] C. Wolf and D. Doermann. Binarization of low quality text using a markov random field model. *Proceedings of International Conference on Pattern Recognition*, 2002. 1, 4

[19] Y. Yang and H. Yan. An adaptive logical method for binarization of degraded document images. *Pattern Recognition*, pages 787–807, 2000. 1

[20] M. Yasuda, J. Ohkubo, and K. Tanaka. Digital image inpainting based on markov random field. *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce*, 2:747–752, 2005. 2

[21] J.-Y. Yoo, M.-K. Kim, S. Y. Han, and Y.-B. Kwon. Line removal and restoration of handwritten characters on the form documents. *Proceedings of the 4th International Conference on Document Analysis and Recognition*, pages 128–131, 1997. 2