A Two-Step Adaptive Descreening Method for Scanned Halftone Image

Fei Chen[1], Shutao Li[1], Le Xu[2], Bin Sun[1], and Jun Sun[3]

[1]College of Electrical and Information Engineering, Hunan University, Changsha, China, 410082
[2]School of Automation, Southeast University, Nanjing, China, 210096
[3]Information Department, Fujitsu Research and Development Center, Beijing, China, 100025
cs_fei9009@hotmail.com, shutao_li@hnu.edu.cn, 522608533@qq.com, sunbinxs@126.com, sunjun@cn.fujitsu.com

Abstract. Halftoning is a necessary technique for electrophotographic printers to print continuous tone images. Scanned images obtained from such printed hard copies are corrupted by screen like artifacts called halftone patterns. Descreening aims to recover high quality continuous tone image from the scanned image. In this paper, a two-step descreening method is proposed to remove screen like artifacts adaptively. Firstly, an Extreme Learning Machine (ELM) based halftone image classification scheme is introduced to categorize the scanned images into different resolutions. Then in the halftone pattern removal step, patch similarity based smoothing filtering and nonlinear enhancement are combined to remove halftone patterns and preserve the image quality. Experiments demonstrate that the proposed method removes halftone patterns effectively, while preserving more details and recovering cleaner smoothing regions.

Keywords: Scanned image, descreening, halftone, adaptive parameters.

1 Introduction

Currently, most electrophotographic (EP) printers adopt a halftone technique to approximate the original contone image with a binary halftone image. However, annoying halftone patterns often appear in scanned images of such printed hard copies. These patterns decrease the aesthetic quality of scanned images. Moreover, the periodic halftone patterns introduced by clustered dots halftoning[1], the most commonly used halftoning technique in current EP printers, may produce Moiré effect in hard copies if the scanned images are reprinted[2].

Several methods[3]-[6] recover contone images with details and sharp edges from binary halftone images. Nevertheless, these methods can only work on binary halftone images and they are not suitable to descreen the scanned halftone images because scanned images are grayscale. Some other descreening methods are designed for the scanned halftone image. Intuitively, the simplest way of descreening is to perform low-pass filtering on the scanned image. However, these

S. Li et al. (Eds.): CCPR 2014, Part II, CCIS 484, pp. 97–106, 2014.

filtering approaches have difficulties in striking a balance between detail preservation and halftone patterns removal. Siddiqui *et al.*[2][7] introduced two descreening methods for scanned halftone images. The training based method[2] contains two steps: basic prediction of contone image, and modified SUSAN filtering based on the predicted version. The authors adopt two schemes to predict the basic contone image, one is simply the Gaussian filtering; the other is resolution synthesis based denoising. And the basic image is used to guide the modified SUSAN filtering to obtain the final descreened image. The results of the method have sharp edges. However, it cannot recover high quality smooth regions. And in the resolution synthesis based denoising, different resolution sample image pairs need to be collected and registrated for training. The second method used local gradient information to estimate contone value[7]. Although it has high computational efficiency, this method cannot remove the halftone patterns along edges effectively.

Most of the above methods for scanned images did not pay much attention to the printing and scanning process. In fact, the halftone patterns in the scanned images vary a lot at different scanning resolutions, but most of these methods are not able to select parameters adaptively. This may lead to detail loss and blurred edges in the recovered contone images at some resolutions.

In this paper, a two-step method is proposed to descreen the scanned image adaptively. In the first step, a halftone image classification is proposed to classify the scanned images into different scanning resolutions. The Local Binary Pattern (LBP) feature is extracted and the Extreme Learning Machine (ELM) is used for classification. And in the second step, contone images with high quality are recovered by an adaptive halftone pattern removal algorithm, whose parameters are determined by the classification results. A patch similarity based smoothing filter is used to remove the halftone patterns, and a nonlinear enhancement is followed to improve the details of the recovered contone images.

2 Adaptive Scanned Image Descreening

Fig.1 is an overview of the proposed descreening method. It consists of two steps: scanning resolution classification, and adaptive halftone pattern removal. The classification step focuses on distinguishing different scanning resolutions of scanned halftone images. Scanned images with known resolutions make a training dataset. The LBP features of all the scanned images in the training dataset is extracted to train an ELM classifier. For the input scanned image with arbitrary resolution, the ELM classifier is used to predict its scanning resolution. In the adaptive halftone pattern removal step, a patch similarity based smoothing filtering is used to remove halftone patterns of the scanned image, and a nonlinear enhancement is followed to improve the sharpness and contrast of the recovered contone image. The parameters of the patch similarity based smoothing filter and the nonlinear enhancement are adaptively selected by the predicted resolution of the input image.

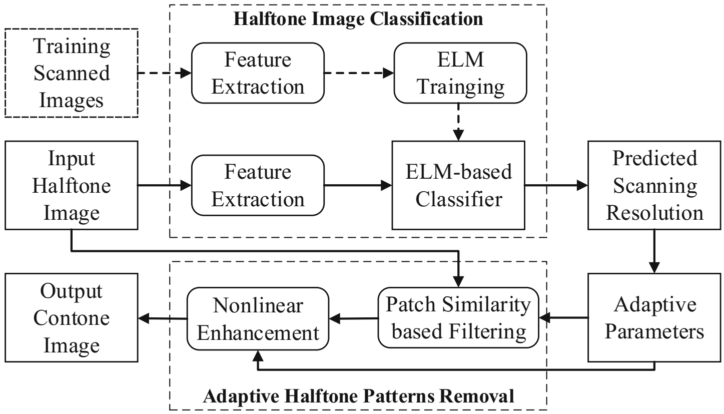


Fig. 1. Overview of the proposed descreening method

2.1 Halftone Image Classification

As shown in Fig.2, the forms of halftone patterns vary with the scanning resolutions, but are seldom affected by the different printing setups because the lines per inch(lpi) is a constant in the same printer. In order to achieve better descreening performance, an ELM based classification strategy is proposed to learning the scanning resolution of the scanned image in the first step.

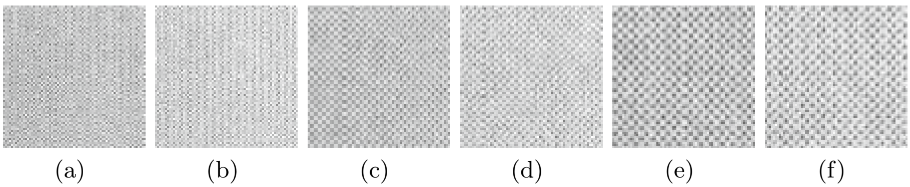


Fig. 2. Halftone patterns printed and scanned at different resolutions. (a) Printed at 300 dpi and scanned at 200 dpi. (b) Printed at 600 dpi and scanned at 200 dpi. (c) Printed at 300 dpi and scanned at 300 dpi. (d) Printed at 600 dpi and scanned at 300 dpi. (e) Printed at 300 dpi and scanned at 400 dpi. (f) Printed at 600 dpi and scanned at 400 dpi.

a) LBP Feature Extraction

The LBP operator[8] has been proved to be an effective texture descriptor. And a widely used extension to the original operator is the so-called uniform patterns represented by $LBP_{P,R}^{u2}$, where P is the number of neighbors and R is the radius of the neighborhood. Its definition is the binary pattern which contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. For example, the patterns 00000000, 01000000, and 00111000 are uniform patterns, while 00101000, 10111000 are non-uniform patterns. In practice, we use uniform patterns with neighborhoods of $P = 8, R = 1$ computed in the whole scanned image so that the LBP feature histogram has 59 bins.

b) Classification Using ELM

For classification, ELMs[9] are selected to train the classifier because of its user-friendly and efficient. Each training example is represented by (x_i, y_i), where i is the serial number of training images, x_i is a 59-dimensional feature vector of scanned image, and y_i is the associated scanning resolution, for three-class, 200dpi, 300dpi and 400dpi respectively. Sigmoid function is selected as the activation function of ELM. Through tuning the numbers of hidden nodes, we can train the ELM to get the optimal network parameters. And then these parameters are used to make up a classifier for scanned images classification. The related training and testing results are shown in the Section 3.

2.2 Adaptive Halftone Pattern Removal

a) Adaptive Patch Similarity Based Smoothing Filtering

In the proposed descreening algorithm, an adaptive patch similarity based filtering strategy is used to remove halftone patterns in the scanned image. The adaptive patch similarity based filter can be written as follows:

$$v(i) = \frac{1}{z(i)} \sum_{\theta \in \Theta} \sum_{r=1}^{R} w(i, i_{\theta,r}) h(i_{\theta,r}) \tag{1}$$

where the normalization factor $z(i)$ is described in (2), $w(i, i_{\theta,r})$ is the Gaussian kernel which is calculated with (3),

$$z(i) = \sum_{\theta \in \Theta} \sum_{r=1}^{R} w(i, i_{\theta,r}) \tag{2}$$

$$w(i, i_{\theta,r}) = \frac{1}{z_w} exp\left(-\frac{d^{patch}(i, i_{\theta,r})}{2\sigma^2}\right) \tag{3}$$

where z_w is a normalization factor, σ is a scale factor which is determined by the predicted resolution, $d^{patch}(i, i_{\theta,r})$ measures the similarity of two patches centered at pixel i and $i_{\theta,r}$ respectively, θ is the directions of patch searching, r is the searching steps, R is the searching radius.

To increase the computing efficiency, the patches along the Minimal Hop Path(MHP)[10] are searched in this step. The MHP is the path with the minimal number of hops connecting two patches, and we only consider MHPs along 8 connectivity discrete directions $\left(\Theta = \{\frac{i\pi}{4} | i = 1, 2, ..., 8\}\right)$. Considering one MHP direction, $d^{patch}(i, i_{\theta,r})$ is shown as Eq. (4).

$$d^{patch}(i, i_{\theta,r}) \approx \sum_{t=1}^{r} \|N(x_t) - N(x_{t-1})\| \tag{4}$$

where $\theta \in \Theta$, $N(x)$ is the patch centered at pixel x , r is the hop numbers. Eq. (4) can further equal to

$$d^{patch}(i, i_{\theta,r}) \approx d^{patch}(i, i_{\theta,r_1}) + \|N(x_r) - N(x_{r-1})\| \tag{5}$$

It indicates the $d^{patch}(i, i_{\theta,r})$ can be computed progressively: we can first compute 1-hop path distance and then propagate it to 2-hop, 3-hop, and so on. In practice, we use patch size 7×7 and window size 13×13, because smaller size cannot involve enough similarity information, and too large size will decrease the efficiency.

b) Nonlinear Enhancement

Although the adaptive patch similarity based filtering can remove the halftone patterns effectively, it cannot reduce nonlinear effects introduced in the printing and scanning process, such as contrast reduction. Therefore, a nonlinear enhancement is proposed to improve the contrast and sharpen the edges.

Let the output v of adaptive patch similarity based filtering be an input image, a Gaussian blurring filter is first performed on v to get the basic base layer v_b of v , shown as Eq. (6):

$$v_b(x, y) = \sum_{i,j} G(i, j)v(x + i, y + j) \tag{6}$$

To avoid strong edges being blurred in the decomposition process, the basic base layer v_b is refined by the guided filtering[11] with the original image v serving as the joint image:

$$\hat{v}_b = \mathit{guidedfilter}(v_b, v) \tag{7}$$

And the detail layer v_d is obtained by the following equation:

$$v_d = v - \hat{v}_b \tag{8}$$

In order to enhance the details of the recovered contone image, a nonlinear enhancement is performed on the detail layer v_d. Eq. 9 formalizes the process:

$$\hat{v}_d = HP(s \times Th(v_d)) \tag{9}$$

where $\hat{v}_d$ is the enhanced detail layer, HP is the high-pass filtering process, s is a scaling constant and $Th(x)$ is the following nonlinear function:

$$Th(x) = \begin{cases} c \cdot x_{max}, & if \quad x > c \cdot x_{max} \\ x, & if \quad c \cdot x_{max} \geq x \geq -c \cdot x_{max} \\ -c \cdot x_{max}, & if \quad x < -c \cdot x_{max} \end{cases} \tag{10}$$

where c is the clipping constant ranging between 0 and 1. With the enhanced detail layer $\hat{v}_d$, the output contone image can be obtained as follows:

$$\hat{I}_o = \hat{v}_b + \hat{v}_d \tag{11}$$

Finally, Eq. 12 is used to improve the image contrast to get the final recovered contone image I_o of our descreening method.

$$I_o = \frac{\hat{I}_o - min(\hat{I}_o)}{max(\hat{I}_o) - min(\hat{I}_o)} \tag{12}$$

In this enhancement process, the values of c and s are adaptively selected in an experiential parameter table by the classification results, and the low-pass Gaussian filter kernel G and the high-pass filter kernel HP are given by

$$G = \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix} /256,\ HP = \begin{bmatrix} 1 & -4 & 6 & -4 & 1 \\ -4 & 16 & -24 & 16 & -4 \\ 6 & -24 & 36 & -24 & 6 \\ -4 & 16 & -24 & 16 & -4 \\ 1 & -4 & 6 & -4 & 1 \end{bmatrix} /256.$$

3 Experimental Results

In order to evaluate performance of the proposed method on real scanned images, we print and scan the 60 images from the Berkeley Segmentation Database. Each original contone image is printed by printer RICOH Aficio MP4500 at 300 and 600 dpi respectively. And every hard copy is scanned by scanner Fujitsu fi-6130 at 200, 300 and 400 dpi respectively. So for each scanning resolution, 120 images are used to test the proposed method.

3.1 Results of Halftone Image Classification

To evaluate the proposed halftone image classification algorithm, the whole dataset listed in Table 1 is equally and randomly partitioned into the training set and test set. Fig. 3 shows the classification accuracies with different numbers of hidden nodes. The result is the average of 20 random experiments. From Fig. 3, it can be found that the accuracy on the training set increases and reaches 100% when the number of hidden nodes is more than 25. The accuracy on the test set increases to 99.4% with about 31 hidden nodes, and decreases gradually if the number of hidden nodes are more than 50, due to the ELM classifier overfitting the training set. In our experiment, the number of hidden nodes is selected to 31.

3.2 Comparison with Existing Methods

The proposed descreening method is compared with three existing methods: training-based descreening using Gaussian filter(TBD-I); training-based descreening using resolution synthesis(TBD-II) and hardware friendly descreening(HFD). The software of the training-based descreening is available on the website. The HFD method is implemented in Matlab. Table 2 shows the parameters of TBD-I, TBD-II and HFD used for the experiments on images scanned at 300 dpi.

Table 1. Experimental dataset

Class		Numbers of image	Train Set	Test Set
Original Contone Image		120	60	60
Scanned Halftone Image	Scanned at 200 dpi	120	60	60
	Scanned at 300 dpi	120	60	60
	Scanned at 400 dpi	120	60	60

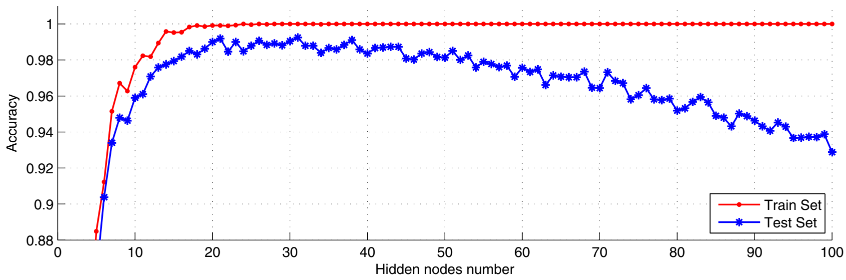


Fig. 3. Accuracy of halftone image classification

Table 2. Parameter settings of the existing methods used in the experiments

Method	Parameter	Value
TBD-I	Radius for Gaussian filter	3
	Scale factor for Gaussian filter	2.5
	Radius for modified SUSAN filter	3
	Space scale factor	2.5
	Brightness scale factor	21
TBD-II	Delta	2.2
	Radius for modified SUSAN	3
	Space scale factor	2.5
	Brightness scale factor	21
HFD	Filter radius	3
	Sharpness level	0

The parameter values of these three methods are selected as reported in the paper. And the parameter values of the proposed method are listed in Table 3.

Results of different descreening methods are compared in Fig. 4. Fig. 4(a) is the original scanned image. The descreening results of TBD-I, TBD-II, HFD and the proposed algorithm are shown from Fig. 4(b) to Fig. 4(e) respectively. From Figs. 4(b) and 4(c), the TBD method removes the halftone patterns very

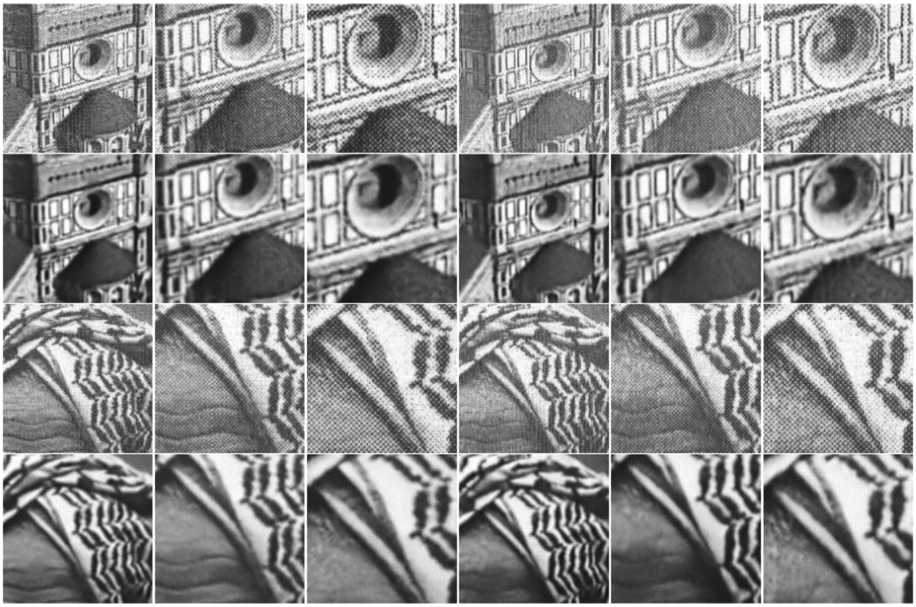
Table 3. Parameter settings of the proposed methods used in the experiments

Step	Parameter	Scanned at 200 dpi	Scanned at 300 dpi	Scanned at 400 dpi
Halftone Image Classification	Hidden nodes	31	31	31
Halftone Pattern Removal	Scale factor σ	22	26	35
	Clipped constant c	0.1	0.2	0.3
	Scaling constant s	1	2	3

(a) (b) (c) (d) (e)

Fig. 4. Close-up views of results of different descreening methods. (a) Original scanned images printed at 300 dpi and scanned at 300 dpi. (b) Results of TBD-I. (c) Results of TBD-II. (d) Results of HFD. (e) Results of the proposed method.

well. But the edges produced by the TBD-I method are blurred as shown in Fig. 4(b). The TBD-II method preserves sharp edges, but however, the smooth regions of the results are noisy. In Fig. 4(d), the HFD method cannot remove halftone patterns along the edges. As shown in Fig. 4(e), it can be found that the proposed method removes halftone patterns in smooth regions as well as along edges with improving the image contrast. The recovered edges of the proposed method are clearer and sharper than that of TBD-I method. Compared with



(a) (b) (c) (d) (e) (f)

Fig. 5. Results of the proposed method on scanned image with different printing and scanning resolutions. (a) The original scanned images printed at 300dpi and scanned at 200dpi and corresponding descreened images. (b) Printed at 300 dpi and scanned at 300 dpi and corresponding descreened images. (c) Printed at 300dpi and scanned at 400 dpi (d) Printed at 600 dpi and scanned at 200 dpi. (e) Printed at 600 dpi and scanned at 300 dpi. (f) Printed at 600 dpi and scanned at 400 dpi.

TBD-II method, the proposed method produces much cleaner smoothing regions with better visual effect.

3.3 Different Printing and Scanning Setup

Fig. 5 shows results of the proposed method on scanned images with different printing and scanning resolutions. These results are produced using the same parameters shown in Table 3. From Fig. 5, we can see the proposed method produces very good results for different printing and scanning resolutions. Although halftone patterns become more prominent with the increasing scanning resolution, the details of our result can be still preserved well with only a little decreasing.

4 Conclusions

In this paper, a two-step descreening method is proposed to recover high quality contone image for the scanned halftone image. In the first step, an ELM classifier is used to learn the scanning resolution of the input image by using the LBP features. The predicted resolution is then used for the adaptive parameter selection of the next step. In the second step, the adaptive halftone pattern removal is performed by combining a patch similarity based smoothing filter and a nonlinear enhancement. Experiments on real scanned images show that the proposed descreening method can recover high quality contone images with both sharp edges and clean smooth regions.

Acknowledgement. This work was supported in part by the National Natural Science Foundation of China under Grant (No. 61172161), the National Natural Science Foundation for Distinguished Young Scholars of China under Grant (No. 61325007).

References

1. Sullivan, J., Ray, L., Miller, R.L.: Design of Minimum Visual Modulation Halftone Patterns. IEEE Trans. Syst., Man, Cybern. 21(1), 33–38 (1991)
2. Siddiqui, H., Bouman, C.A.: Training-Based Descreening. IEEE Trans. Image Process. 16(3), 789–802 (2007)
3. Stevenson, R.L.: Inverse halftoning via MAP estimation. IEEE Trans. Image Process. 4(4), 486–498 (1997)
4. Chang, P., Yu, C., Lee, T.: Hybrid LMS-MMSE Inverse Halftoning Technique. IEEE Trans. Image Process. 10(1), 95–103 (2001)
5. Chen, L., Hang, H.: An Adaptive Inverse Halftoning Algorithm. IEEE Trans. Image Process. 6(8), 1202–1209 (1997)
6. Liu, Y., Guo, J., Lee, J.: Inverse Halftoning Based on the Bayesian Theorem. IEEE Trans. Image Process. 20(4), 1077–1084 (2011)
7. Siddiqui, H., Bouman, C.A.: Hardware-Friendly Descreening. IEEE Trans. Image Process. 19(3), 746–757 (2010)
8. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. IEEE Trans. Pattern Anal. Mach. Intell. 24(7), 971–987 (2002)
9. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: theory and applications. Neurocomputing 70, 489–501 (2006)
10. Chen, X., Kang, S.B., Yang, J., Yu, J.: Fast Patch-Based Denoising Using Approximated Patch Geodesic Paths. In: Proc. IEEE Conf. Comput. Vision Pattern Recog., pp.1211–1218 (2013)
11. He, K., Sun, J., Tang, X.: Guided Image Filtering. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 1–14. Springer, Heidelberg (2010)