A Line Drawings Degradation Model for Performance Characterization

[1]Jian Zhai, [2]Liu Wenyin, [3]Dov Dori, [1]Qing Li

[1]*Dept. of Computer Engineering and Information Technology;* [2]*Dept of Computer Science*
City University of Hong Kong, {jian.zhai@student.;csliuwy@;itqli@}cityu.edu.hk
[3]*Fculty of Industrial Engineering and Management, Technion, Israel, dori@ie.technion.ac.il*

Abstract

Line detection algorithms constitute the basis for technical document analysis and recognition. The performance of these algorithms decreases as the quality of the documents degrades. To test the robustness of line detection algorithms under noisy circumstance, we propose a document degradation mode, which simulates noise types that drawings may undergo during their production, storage, photocopying, or scanning. Using our model, a series of document images at various noise levels and types can be generated for testing the performance of line detection algorithms. To illustrate that our model is consistent with real world noise types, we validated the method by applying it to three line recognition algorithms.

1. Introduction

A line drawings degradation model is needed for two purposes. First, we can use the model to generate noisy line drawing images in order to test the robustness of line detection algorithms, which are the foundation of technical document analysis and recognition. If we use real world documents, we need to obtain their ground truth information manually. This is a labor-intensive and error-prone procedure, which cannot achieve accurate results. A synthesized degraded document has advantages over real world documents for testing, because its ground truth prior to the degradation procedure is known with certainty. Second, the model makes it possible to control the intensity and density of quality degradation. In other

words, the level and types of noise applied to it can be controlled with precision.

Many modeling degradation procedures have been proposed. Loce and Lama modeled the perturbation introduced due to mechanical disturbances [1]. Baird discussed a ten-parameter model for character degradation [2]. Kanungo et al. proposed a nonlinear global and local document degradation model [3], which was later enriched and validated [4].

We propose an approach to simulating the effect of real-world degradation types by mimicking the outcomes of four different sources of noise. It is usually very difficulty to validate a degradation model since noises are randomly distributed and direct pixel to pixel comparison is not effective. In order to test whether the degradation model can describe real world noise types well, we introduce a black box method for testing the degree of degradation. Our experiments show that the proposed degradation model is valid and can therefore be used to synthesize images for evaluating line detection algorithms. The results are close enough to those obtained using real world images, but the high cost of manual processing required to obtain the drawings' ground-truth information is saved.

2. Line Drawings Degradation Model

In our degradation model, ground truth is needed to describe the characteristics of all the lines in the drawing. We convert the ground truth file into a bitmap document, to which the four types of noise will be added. The procedure is shown in Figure 1.
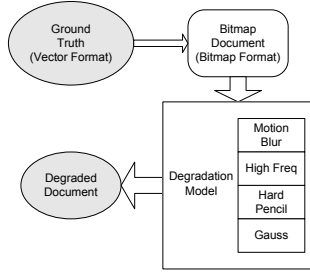
Figure 1. Degradation procedure

Next, we discuss each one of the four types of noise in our model. For each noise type, we introduce its physical cause, mathematical model, and degradation level parameter. Finally, For each noise type we show three sample images at three different noise levels.

2.1. Gaussian Noise

Gaussian noise, also known as white noise, is caused by acoustical or electrical disturbances, and for which the intensity is the same at all frequencies within a given band. It is most likely to be imparted into documents when they are printed or scanned. We add Gaussian noise to a document as described below.

First, for each point/pixel, we generate a random variable (r.v.):

$$R = \frac{rand() \times L\_MAX}{RAND\_MAX} \times 6 \qquad (1)$$

where, $RAND\_MAX$ represents the maximum value the function $rand()$ may generate. L is a threshold ranging from 0 to L_MAX. If $R < L$, a normally distributed r.v. is generated from a uniform distributed r.v. using the Lindeberg-Levy Theory:

$$Z_{12} = \sum_{k=1}^{12} Y_k - 6 \qquad (2)$$

$\{Y_k, 1 \le k \le 12\}$ is a 12-dimensional uniform random variable $U[0,1]$ of independently identical distribution. L indicates the degradation level and is used to judge whether a pixel should be added with Gaussian noise. $L\_MAX = 10$ means that 1/6 of the points are added with Gaussian noise, while $L = 0$ means no degradation. Figure 2 shows the document under Gaussian noise with $L = 0$, 5, and 10, respectively, ordered from left to right.

Figure 2. The effect of a Gaussian noise

2.2. High Frequency Noise

High frequency noise occurs at the high frequency region of an image. In line-intensive documents, the high frequency regions are found along the edges of lines. High frequency noise is most likely to be caused when the lines are drawn or printed on a rough plane. Combined with physical faults of the drawing tools, the edges of lines become ragged.

We use convolution for simulating this type of noise. The high frequency noise is generated as follows. First, we generate a random variable R to determine whether point G_{ij} in the original document should be replaced with a noisy point P_{ij}, using the following formula:

$$R = \frac{rand() \times L\_MAX}{RAND\_MAX} \qquad (3)$$

Using a threshold L, ranging from 0 to L_MAX, if $R < L$, we prepare a convolution kernel K with size $w=L/4+2$. The value of each element in K is randomly generated. Then, we use K to convolute the whole document, using the following formula to calculate the value of each point:

$$p_{ij} = \left(\sum_{m=-w/2}^{(w-1)/2} \sum_{n=-w/2}^{(w-1)/2} k_{ij} g_{mn} \right) \Big/ \left(\sum_{m=-w/2}^{(w-1)/2} \sum_{n=-w/2}^{(w-1)/2} k_{ij} \right) \qquad (4)$$

where, k_{ij} is the (i,j) element in matrix K, with $i=1+m+w/2$ and $j=1+n+w/2$. A matrix G maintains the pixel value of an area in the document, with point g_{ij} in its center. g_{mn} is the (m, n)-th element in G. In the above formula, we divide the convolution result by the sum of all elements in K to map the result into [0, 255].

In this procedure, we define L as the degradation level with $L\_MAX = 10$. $L = 0$ means that this type of

degradation does not exist, while $L = 10$ means that the line will be degraded to a depth of 4 pixels at most.

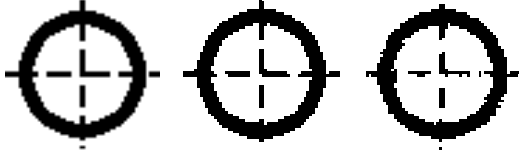Figure 3 shows the document under high frequency noise when $L = 0$, 5, and 10, respectively.

Figure 3. The effect of high frequency noise

2.3. Hard Pencil Noise

Many documents use hard pencils or similar drafting tools. Due to the paper's texture, when the pencil draws on the surface of the paper, some places remain untouched, leaving tiny white gaps across the black lines. We stimulate this degradation by a scan line algorithm. The document is scanned by a line expressed as:

$$x + y = k, k = 0,1,...,w + h \qquad (5)$$

where w is the width of the picture and h is its height.

For each point g_{ij} which is on the scan line and is not a point in a background (white) color, we generate a random variable

$$R = \frac{rand() \times L\_MAX}{RAND\_MAX} \times 13 \qquad (6)$$

Using a threshold L, ranging from 0 to L_MAX, if $R < L$, a random number w between 0 and $(L+5)/3$ is generated. A w-pixel-long white line is drawn at the same direction as the scan line. If this white line extends out of the black area, its tail is cut off, effectively ignored.
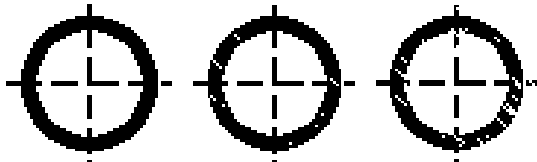

Figure 4. The effect of hard pencil noise

As L, the degradation level, increases, more and longer white lines appear on the black area, making the noise

stronger. Figure 4 shows the document under hard pencil noise when $L = 0$, 5, and 10, respectively.

2.4. Motion Blur Noise

Motion blur noise often occurs when the document is disturbed and moved during its scanning or snapping. The direction of motion is selected randomly. Each pixel of the original document is divided by L to get a document M. M is then moved along the selected direction by i pixels, where i is an integer ranging from $-L/2$ to $L/2$. Each time, M_i is added to M. Any pixel value larger than 255 is set to 255. The result is the motion-blurred document M'. L represents the noise level. We limit L to [0, 10]. The blur effect, with $L = 0$, 5, and 10, respectively, is shown in Figure 5.

Figure 5. The effect of motion blur noise

2.5. Drawing Synthesis

In the real world, some subset of the above four types of noise is applied to an image in some order. The principle we follow is that noise introduced later affects the previous noise level, since that has already become part of the document. For example, if we add motion blur noise after hard pencil noise, the white short lines appearing within black areas (as hard pencil noise) may be weaker, or even covered by the motion blur noise. According to this principle, we arrange the four types of noises in the following order: We apply Motion Blur Noise first, which is followed by High Frequency Noise, then Hard Pencil Noise, and finally Gaussian Noise.

We use the same degradation level for all types of noise so we obtain the final document at that degradation level.

3. Validation

3.1. Degradation Level Estimating

Estimating the degradation level of a given document is the most important step in the validation. First, we must obtain the ground truth of a given image (D_0) manually. $D(x, y)$ is the pixel value at point (x, y). Then we use the ground truth to generate a new bitmap document (D_1) without any degradation and obtain $D' = D_0 - D_1$ as the intensity of the degradation noise of the original document. We also generate a new most noisy document D_{10}, with a noise level $L=10$, and compute $D'' = D_0 - D_{10}$. The following formula is used to calculate the estimated degradation level (L_{est}) of D_0:

$$L_{est} = \frac{\sum_{x,y} |D'(x,y)| - \sum_{x,y} |D''(x,y)|}{\sum_{x,y} |D''(x,y)|} \times 10 \qquad (7)$$

3.1. Black Box Testing

The black box testing method for validating the degradation model is as follows. For any given scanned or synthesized degraded document, using the above formula, we first compute the document's L_{est}, its estimated degradation level. We then apply our degradation model to synthesize a degraded document at the level L_{est}. Finally, we compute the noise difference between the original degraded document and the synthesized degraded document. If the difference is small enough, as specified in the algorithm below, we say that the model is valid.

---

1. Estimate the degradation level L_{est} of a real-life document D_0.
2. Manually get the ground truth D of D_0.
3. Generate a degraded document D_1 from D at degradation level L_{est}.
4. Compute $\varepsilon = |\sum |D_0 - D| - \sum |D_1 - D||$.
5. If $\varepsilon \le 0.1 \times \sum D$, accept the model;
   Else, reject the model.

In our model, $L\_MAX=10$, so the degraded document can be at one of 10 levels. Hence, the discrimination granularity of degradation levels in our model is 1/10. That is why we use 0.1 in Step 5.

4. Experiments and Results

4.1. Validation of Our Model

To validate our degradation model, we select three real-life line drawings with various sizes and degradation levels. Their raster images are shown in Figure 6.
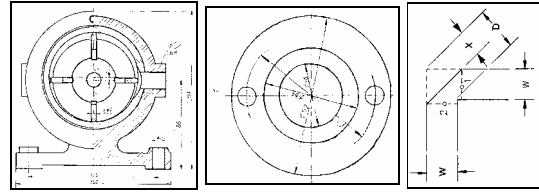

Figure 6. The scanned line drawings used in the experiment

We manually obtained their ground truths and calculated the ε of each model using the method described in Section 3. The results are shown in Table 1.

Table 1. Validation results

Doc Size	L_{est}	ε	$\sum D$	Result
3510×2550	4	20425	408503	Accept
748×646	7	1174	32872	Accept
512×512	2	787	13802	Accept

As can be seen from Table 1, the ε of each drawing is smaller than $0.1 \times \sum D$. The results therefore conform with our expectation that they all pass the validation and our degradation model indeed can well simulate real world drawings.

4.1. Performance Evaluation

Using the ground truth obtained from [5], we generated the images with degradation levels ranging from 0 to 10 for testing the three line recognition algorithms described in [6], [7], and [8]. The evaluation tool described in [9] was

used to obtain the performance metrics *VRI* [9]. Table 2 presents the result.

Table 2. Performance metrics of the three algorithms at different degradation levels

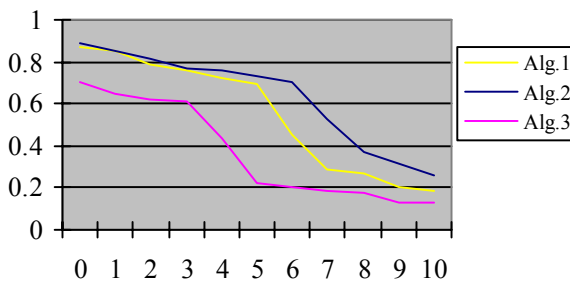| L	0	1	2	3	4	5	6	7	8	9	10
1	.873	.834	.796	.755	.719	.699	.458	.287	.270	.200	.188
2	.892	.852	.819	.771	.758	.734	.703	.527	.370	.313	.257
3	.701	.648	.621	.614	.439	.219	.208	.189	.173	.132	.128

Figure 7. Performance curves of the three algorithms

Figure 7 clearly shows that the performance curves of all three algorithms descend as the document's degradation level increases. Since Algorithm 2's curve is always higher than the other two, it is determined as the best algorithm with respect to robustness to noise generated using our degradation model.

5. Conclusion

We have presented an elaborate line drawings degradation model, which accounts for four types of degradation: Gaussian, high frequency, hard pencil, and motion blur, with a uniform definition of the degradation level. The four types of degradation noise can be added to a ground truth sequentially and generate synthesized degraded images at different levels. The model is validated by a black-box testing method. Using our validated degradation model, a series of images can be generated with increasing degradation levels. Applying three line detection algorithms we have shown that the performance of line recognition algorithms decreases as the degradation level increases. We can judge effectively which algorithm is the best by comparing their performance curves at different degradation levels.

References

[1]. R. Loce and W. Lama, "Half-tone Banding Due to Vibrations in a Xerographic Image Bar Printer", *Journal of Imaging Technology*, 1990, 16(1):6-11.

[2]. H. S. Baird, "Document Image Defect Models", *Structured Document Image Analysis*, Springer-Verlag, New York, 1992.

[3]. T. Kanungo, R. M. Haralick, I. Phillips, "Global and Local Document Degradation Models", *Proc. of ICDAR93*, 1993, pp. 730-734.

[4]. T. Kanungo, R. M. Haralick, H. S. Baird, W. Stutzle, and D. Madigan, "Document Degradation Models: Parameter Estimation and Model Validation", *Proc. of Workshop on Machine Vision and Applications*, Japan, 1994, pp. 552-7.

[5]. Liu W., J. Zhai, and D. Dori, "Extended Summary of the Arc Segmentation Contest", *Lecture Notes in Computer Science*, Vol. 2390, 2002, pp. 343-349. http://www.cs.cityu.edu.hk/~liuwy/ArcContest/arctruth.vec

[6]. Liu W. and D. Dori, "Incremental Arc Segmentation Algorithm and Its Evaluation", *IEEE Trans. on PAMI*, Vol. 20, No. 4, April 1998.

[7]. D. Elliman, "TIF2VEC, An Algorithm for Arc Segmentation in Engineering Drawings", *Lecture Notes in Computer Science*, Vol. 2390, 2002, pp. 350-358.

[8]. http://xmailserver.org/davide.html.

[9]. Liu W. and D. Dori, "A Protocol for Performance Evaluation of Line Detection Algorithms", *Machine Vision and Applications*, Vol. 9, 1996, pp. 240-250.