Parameter Calibration for Synthesizing Realistic-Looking Variability in Offline Handwriting

Wen Cheng[a] and Daniel Lopresti[b]

[ab]Lehigh University, 19 Memorial Dr. West, Bethlehem, USA

ABSTRACT

Being motivated by the widely accepted principle that the more training data we have, the better performance the recognition system has, we conducted experiments asking human subjects to do test on a mixture of real English handwritten textlines and textlines altered from existing handwriting with various distortion degrees. The idea of generating synthetic handwriting is based on a perturbation method by T. Varga and H. Bunke that distorts an entire textline. There are two purposes of our experiments. First, we want to calibrate optimal distortion parameter settings for Varga and Bunke's perturbation model. Second, we intend to compare the effects of parameter settings on different writing styles, block, cursive and mixed. From the preliminary experimental results, we determined appropriate ranges for parameter amplitude, and found that parameter settings should change for different handwriting styles. Once the proper parameter settings are found, we will generate large amount of training and testing sets for building better off-line handwriting recognition systems.

Keywords: Offline handwriting recognition, Perturbation model, Synthetic handwriting, Parameter calibration

1. INTRODUCTION

In the field of pattern recognition, it was widely accepted that size and quality of the training data have critical effect on the accuracy of the recognition system.[1] Inspired by a well known rule of thumb saying that the classifier wins that is trained on the most data, many researchers examined the effect of expanding the size of training data set on the performance of handwriting recognition system. This idea was shown successful through experiments.[2,3] However, those experiments were conducted using true handwriting as training data, which is it is very expensive and time consuming together. One way to solve the problem is to synthesize data from original handwritten samples. It was also experimentally shown that enlarging training set by adding synthetic handwriting data could improve the performance of handwriting recognition systems.[4–7]

There are several ways to generate off-line synthetic handwriting,[4,8–12] e.g., the point correspondence method[13] and template based synthetic handwriting generation.[14] However, most of them are applied to isolated characters. We are working on handwritten textlines as inputs to our handwriting recognition system, so we choose the perturbation model of Varga and Bunke,[15] which will be briefly described in Sec. 3.

As mentioned, in addition to the size of the training data the quality is also important to recognition system's performance. One function of adding synthetic data to a training set is to increases its variability, so the distortion should not be too weak. On the other hand, severely distorted handwriting may bias the recognition system toward unnatural handwriting styles, which can lead to a deterioration of the recognition accuracy. So it is very important to find an optimal distortion parameter range and control the variability of data to ensure that the synthetic handwriting is natural enough. Motivated by this idea, we conducted experiments to determine the appropriate parameter settings for the Varga and Bunk's model, focusing specifically on the "amplitude" parameter, a, to be explained in Sec. 3.

First, we collected handwriting sentences of different writing styles from 15 English native speakers. Then we used the computer software we have developed to alter the handwriting to various degrees. Finally, we show some of the original sentences, along with the others that have been altered to another 21 English native

Further author information: (Send correspondence to Wen Cheng)
Wen Cheng: E-mail: wec206@lehigh.edu, Telephone: 1 610 758 5784
Daniel Lopresti: E-mail: dal9@lehigh.edu, Telephone: 1 610 758 5782

speakers. For each image, the subjects were asked to decide whether it reflected "real" or "altered" handwriting. An "unsure" option was also available. From the opinions collected, we computed a baseline for measuring how likely the subjects think the image was altered handwriting even when it was real. Once we had this baseline, we calibrated appropriate settings for parameters of the perturbation model. We also consider comparisons across writing style.

The structure of the rest of this paper is shown as follows. We first summarize the Varga and Bunke's experiments and their results on finding proper use of these synthetic data. Then we briefly review the mechanism of Varga and Bunke's model in addition to our variations of it, followed by the human subject parameter setting experiments conducted on the English handwriting scanned images. Finally we conclude with a brief summary and a discussion of future work.

2. RELATED WORK

Some previous work has been down to determine appropriate distortion strengths for synthetic handwriting generated by Varga and Bunke's model. In Ref. 5–7, Varga and Bunke conducted writer-independent experiments to see whether the performance of an off-line handwritten text recognizer described in Ref. 16 can be improved by adding synthetically generated texts to the training data. In their experiments, they determined appropriate distortion strengths and recognizer capacity (i.e. the number of Gaussians) by checking whether they could lead to significant improvement of the recognition rate. Generally, their experimental results demonstrated that using such expanded training sets could improve the recognition rates. Particularly, when using a small training set with a large number of writers, two factors affected the performance: the number of Gaussians and distortion strengths. Usually, a larger number of Gaussians could always generate better recognition performance. Furthermore, the larger the training set size was, the larger the number of Gaussians was expected, because there might be danger that unnatural synthetic data present in the expanded training set could degrade the recognition performance. The distortion strengths were divided into 4 levels, very weak, weak, middle, and strong. It was found that the recognition rates were improved the most when distortion strength was weak. It was also shown that significant improvement in the recognition rate was possible even in the case of a large training set provided by many writers.

Our experiments take advantage of user input to calibrate the distortion parameter settings. Beyond the experimental protocol we have developed and the actual settings which we report here, a primary contribution of this work is determining that parameter settings should differ for different writing styles (block, cursive, mixed).

3. MECHANISM OF HANDWRITING SYNTHESIS

The perturbation method generates synthetic textlines from existing handwriting textlines. The basic approach is to use continuous nonlinear functions that control a class of geometrical transformations.

3.1 Underlying Functions

Each transformation is controlled by a continuous nonlinear function, which determines the strength of the considered transformation in various directions. The underlying functions is called *CosineWave*. A CosineWave is the concatenation of n functions, f_1, f_2, ..., f_n, each a cosine half-wave, where

$$f_i(x) = (-1)^i \times a \times \cos(\frac{\pi}{l_i} \times x), l_i > 0. \quad (1)$$

Figure 1 shows an example of CosineWave functions. f_i (separated by vertical line segments in Figure 1) are called *component*. The *length* of component f_i is l_i and its *amplitude* is $|a|$. The amplitude dose not depend on i, i.e. it is the same for all the components. By randomly picking values of l_i and a, arbitrary CosineWave functions can be generated. In our current study, when generating a synthetic textline, we simplified by setting all values of l_i to a common value l, resulting in a sinusoidal waveform of fixed period. We only varied the value of a, while keeping l the same for a certain transformation.

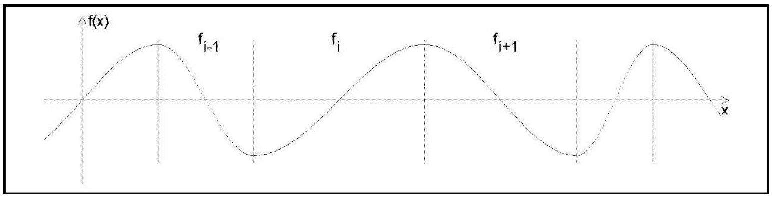

Figure 1: An example of CosineWave functions, which consist of series of piecewise. The picture is from Refs. 15.

3.2 Four Geometrical Transformations

There are four kinds of transformations, shearing, horizontally scaling, vertically scaling and baseline bending. Shearing and vertical scaling depend on defining a straight lower baseline beneath the text. The pixels on the lower baseline are not shifted and are not affected by these two transformations. In the following, the geometrical transformations will be defined and illustrated by figures. The figures are only for illustration purpose and weaker distortions are suggested when being applied to a synthetic training set.

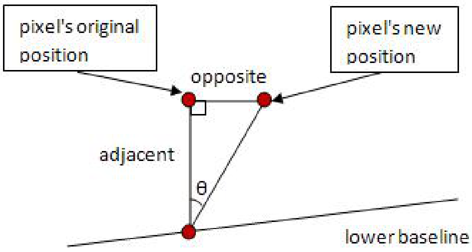

Figure 2: Illustration of shearing a pixel.

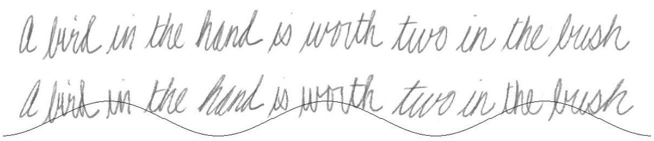

Figure 3: Example of shearing (upper line is original).

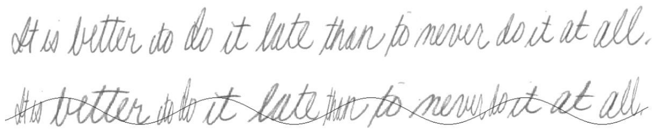

Figure 4: Example of horizontal scaling.

Shearing: For shearing, the underlying function gives the tangent of the shearing angle θ for each text pixel. Consider a particular horizontal location Focusing on one particular text pixel, define its distance above the lower baseline as *adjacent*. Define the pixel's horizontal offset as *opposite*. Then $tan(\theta) = opposite\ adjacent$. An illustration of how the pixel's new position is computed is shown in Figure 2. An example is shown in Figure 3.

Horizontal Scaling: For horizontal scaling, the underlying function defines the shift factor in the horizontal direction for each x coordinate. This transformation is performed by horizontally shifting the pixel columns. An example of this alteration is shown in Figure 4.

Vertical Scaling: The underlying function determines the vertical scaling factor for each x coordinate. Scaling is performed with respect to the lower baseline. For each column of pixels, the part above the lower baseline is scaled bottom-up and the part below the baseline is scaled top-down. We implemented vertical scaling using the *Pixel Mixing method.*[17] An example is shown in Figure 5.

Baseline Bending: The underlying function for each x coordinate defines how much each column of pixels should be shifted vertically. An example is shown in Figure 6.

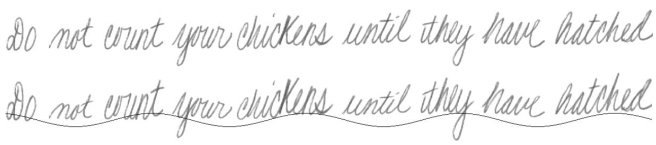

Figure 5: Example of vertical scaling.

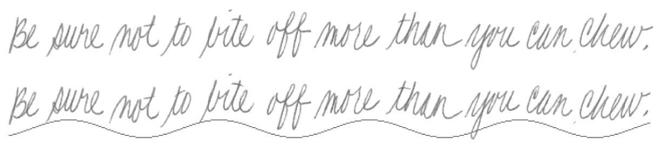

Figure 6: Example of baseline bending.

3.3 Lower Baseline Detection

Accurate lower baseline detection is very important because it determines how a textline will be sheared and vertically scaled. Varga and Bunke's paper did not specify how they detected the lower baseline. We used *second linear regression*[18] for this. The lower baseline and vertical scaling based on that baseline are shown in Figure 7.

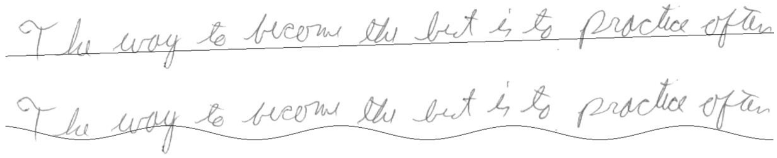

Figure 7: The first line is the textline with lower baseline detected by second linear regression. The second line is the vertically scaled textline based on that lower baseline.

Figure 8: The upper line is horizontally scaled textline without being applied thinning/thickening. The lower line is horizontally scaled textline followed by thinning/thickening.

3.4 Thinning/Thickening

A side effect of horizontal scaling is that the width of strokes also changes, which is particularly obvious when the distortion degree is high. Considering that the human subjects can detect the distortion easily through the width of strokes, we applied post-process to the horizontally scaled handwriting. We investigated several approaches. First we tried the *MB2 Thinning* algorithm.[19] Because this technique is for binary images, it caused strokes to be of uniform thickness and color, as shown in Figure 8. Unfortunately, this made the scaled handwriting looked unnatural to some human subjects.

Finally, morphological erosion and dilation for grayscale images were considered. However, there are two problems caused by this method. First, how to control the number of iterations since too many iterations of thinning/thickening may lead to the opposite effect to the original problem. Second, the eroded part looks lighter and dilated part looks darker.

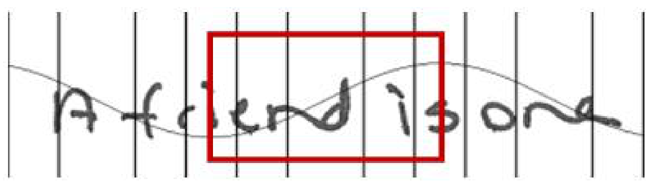

Figure 9: Division of a textline into sections periodically according to the underling functions. The area with the rectangle is half of a period of the Cosine function. There are five sections within that rectangle.

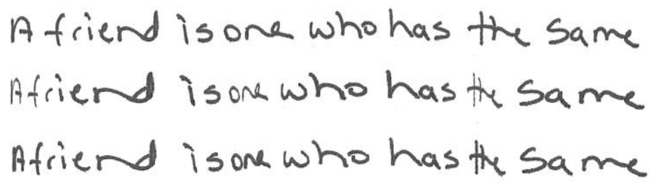

Figure 10: The upper line is the original textline, middle is the horizontally scaled line without post-process, and lower line is after erosion, dilation, and intensity adjustment are applied.

To solve the two problems, we divided each component (as defined before) into five sections, as shown in Figure 9. Within the first and last sections, the function values change subtly so no erosion or dilation is applied. Within the second and fourth sections, the function values change evenly where some iterations are applied. Within the middle section the values change the most dramatically, so many more iterations are applied. Once the area is divided, the average stroke width for each section was computed[20] for both original and distorted textlines. By comparing the stroke widths between each pair of sections the number of iterations that should be applied to the scaled textlines was found.

The width of each section can be computed from three parameters, f, u, and id. f represents the number of fractions a component consists of; u means the number of fractions the unchanging area (the first or fifth section) takes; and id stands for the number of fractions that the second or fourth section takes. Then the lengths of the first (or fifth), second (or fourth), and middle section are $\frac{l \times u}{f}$, $\frac{l \times id}{f}$, and $\frac{l \times (f-u-id)}{f}$, respectively. l is the length of a half cosine wave.

With respect to the second problem, we compute the average image intensity value for each section of both original and distorted image. Then we restored the intensity value of each pixel within a certain section on the distorted image by adding or reducing the the difference between average intensity values of this section. For example, if the average intensity value of section n on original image is x, the average intensity value of the corresponding section m on distorted image is y, and the intensity value of a pixel in m is z, then the new intensity value of this pixel will be z + (x - y) or z - (x - y), depending on whether the function value changes increasingly or decreasingly in section m. The result of our method is shown in Figure 10.

4. EXPERIMENT

The application considered in this paper is the off-line recognition of English handwriting sentences. Our experiments consisted of two parts, the collecting of handwritten sentences (textlines) and of opinions. In the first part, 15 English native speakers as test subjects were asked to write sentences on standard paper. Details are discussed in next subsection. In the second part another group of test subjects were asked whether images of handwriting shown on a computer display were real or altered. The test subjects of the two groups are disjoint.

In the opinion collection part, we recruited 21 English native speakers as test subjects from our university's graduate and undergraduate students, and technical staff. The subjects were from various departments and included both males and females. Before starting the test, they were instructed that they would see a sequence of handwriting images, and that some of the writing was natural and some had been distorted by applying various computer algorithms. We designed a GUI for this experiment. This software showed the handwriting images one by one and for each image there were three radio buttons representing "real", "altered"and "unsure" opinions for subjects to choose. This software not only recorded down each subject's opinions, but also the time spent on every image. Each test usual took one hour and half and the subjects were rewarded at the end.

4.1 Experiment Part 1, and Part 2 Setup

Table 1: 8 distortion strengths for each kind of transformation. The numbers are values of parameter a.

Writing style	# of writers	Textlines collected	# of textlines for baseline	# of synthetic textlines
Block	5	$30 \times 5 = 150$	$(4 \times 4) + (6 \times 1) = 22$	$(26 \times 4) + (24 \times 1) = 128$
Cursive	5	$30 \times 5 = 150$	$(4 \times 4) + (6 \times 1) = 22$	$(26 \times 4) + (24 \times 1) = 128$
Mixed	5	$30 \times 5 = 150$	$(4 \times 4) + (6 \times 1) = 22$	$(26 \times 4) + (24 \times 1) = 128$

We prescreened writers so that were exactly 5 of each of three writing styles. In that way, there were three groups of textlines: 150 cursive sentences, 150 block and 150 mixed. From each group, we selected 22 sentences as baseline (keeping them undistorted), and altered the rest 128 textlines to 8 different degrees with 4 transformations. We tried to choose the range of degrees widely enough so that the weakly distorted handwriting looked natural and strongly distorted one looked obviously unnatural. In addition, each sentence was altered

Table 2: 8 distortion strengths for each kind of transformation. The numbers are values of parameter a.

Transformation type	**Distortion degrees (values of a)**								l
	1	2	3	4	5	6	7	8	
shearing	0.07	0.13	0.19	0.25	0.31	0.37	0.43	0.49	300
3.5ex horizontal scaling	9	13	17	21	25	29	33	37	225
vertical scaling	0.11	0.17	0.23	0.29	0.35	0.41	0.47	0.53	150
baseline bending	1	3	5	7	9	11	13	15	230

only once so that the same sentence never shows up twice in either original or synthetic form in the experiment. Therefore there are 4 synthetic textlines for a certain combination of writing style, transformation, and distortion strength. The distortion strategy is shown in Table 1. Table 2 shows the specific values of 8 distortion degrees (i.e. the 8 values of amplitude a), as well as the chosen value of l for each kind of transformation. Notice that we altered textlines only by changing value of parameter a, while keeping l the same within a specific transformation. As described in Sec. 3, a stands for the amplitude of Cosine Waves, and l stands for the length in pixels of a half period of a cosine wave. Appropriate lengths for l for each transformation were determined by heuristic pretesting. In Figure 11 an example is shown where the textline on top was distorted using the eight different distortion strengths. In the handwritten sentences collection part, we collected 450 textlines (sentences) from 15 English native speakers, and each writer provided 30 sentences, each of 9-12 words. Each sentence was written into a 1.5 inch high by 7.5 inch wide frame on a sheet of paper (a few writers with larger handwriting had 9.5 inch wide frames on landscape-oriented-paper). These were scanned at 300dpi, so typical sentences were about 2000 pixels wide.

Figure 11: Illustration of 8 levels of distortion strengths of shearing transformation. The top line is original and the distortion strengths are in ascent order.

4.2 Experiment Part 2 Results

In this part of the experiment, each human subject was asked to give an opinion (real, altered or unsure) toward each of the 450 textline images. We classified the experimental results into four main categories by transformation. Within each main category we divided the textlines into three sub-categories by writing style. The experiment results are shown in Figure 12, Figure 13, Figure 14, and Figure 15, where each figure describes opinion distributions toward a certain kind of transformation. Each of the three sub-figure describes the opinion distributions toward a certain writing style within a transformation. For example, the first sub-figure in Figure 12 shows the opinion distributions toward cursive sheared handwriting.

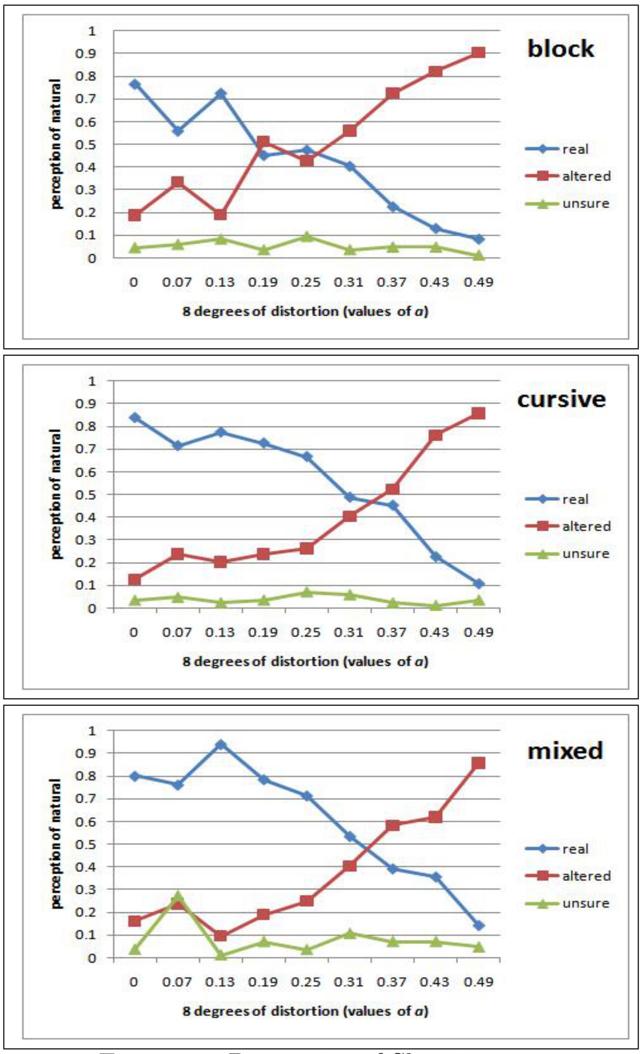

Figure 12: Perception of Shearing.

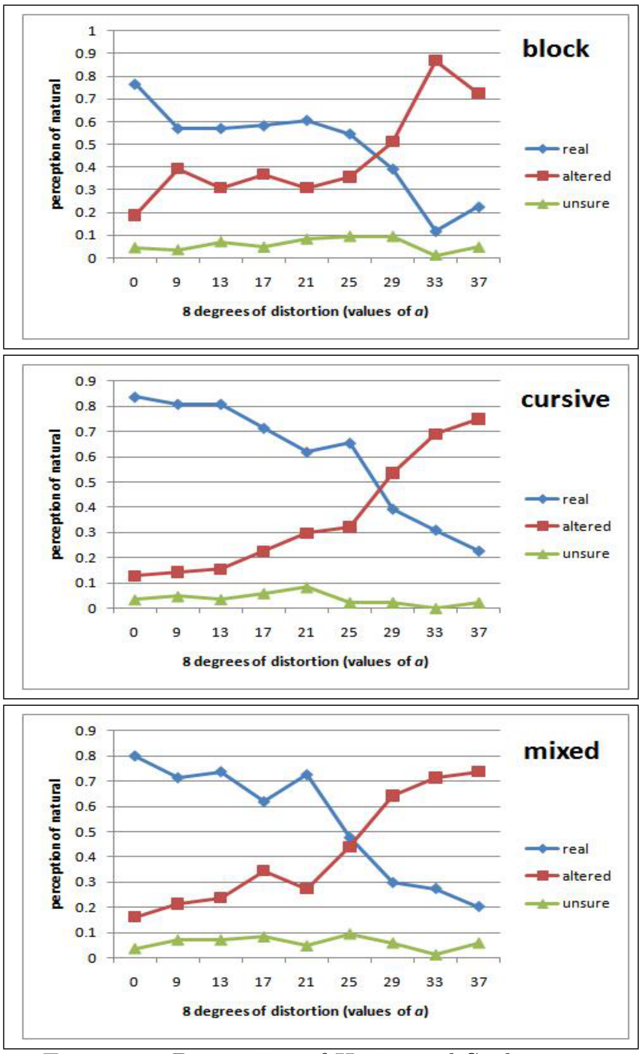

Figure 13: Perception of Horizontal Scaling.

With respect to each sub-figure, the x-axis shows the 8 distortion strengths marked by 8 values of a. The left most value on x-axis is 0, which is the case that the handwriting is not distorted. Note that the values of a are different across transformations. The y-axis shows the proportions of opinions toward a category of handwriting of a certain writing style, transformation and distortion strength. In that way, the sum of the three values for certain distortion strength within each sub-figure (or category) is always **1**. When x=0, The y-values work as baseline for measuring how likely the subjects think the images were altered even when there were real.

As can be seen, subjects felt generally sure of their categorization of a textline as "real" or "altered", with "unsure" seldom chosen, even in the ambiguous moderat-distortion regions. Each sub-figure shows an intersection between the "real" and "altered" lines taking place around the middle of the figure and the trend of "real" and "altered" lines are almost symmetric. Each intersection means there exists one distortion strength for a specific transformation, such that to the left of this strength more synthetic handwriting is thought natural and to the right of which more synthetic handwriting is thought altered (by those who expressed a sure opinion). Table 4 shows approximately 50% crossover points for each combination of transformation and writing style. In the following we discuss the comparison across writing style transformation by transformation based on Table 4.

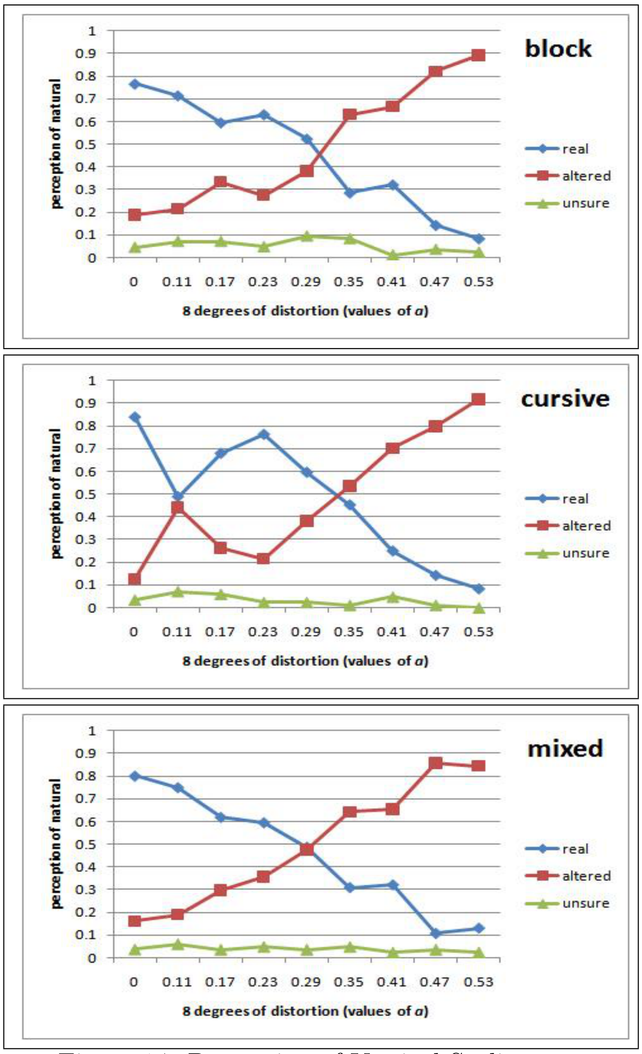

Figure 14: Perception of Vertical Scaling.

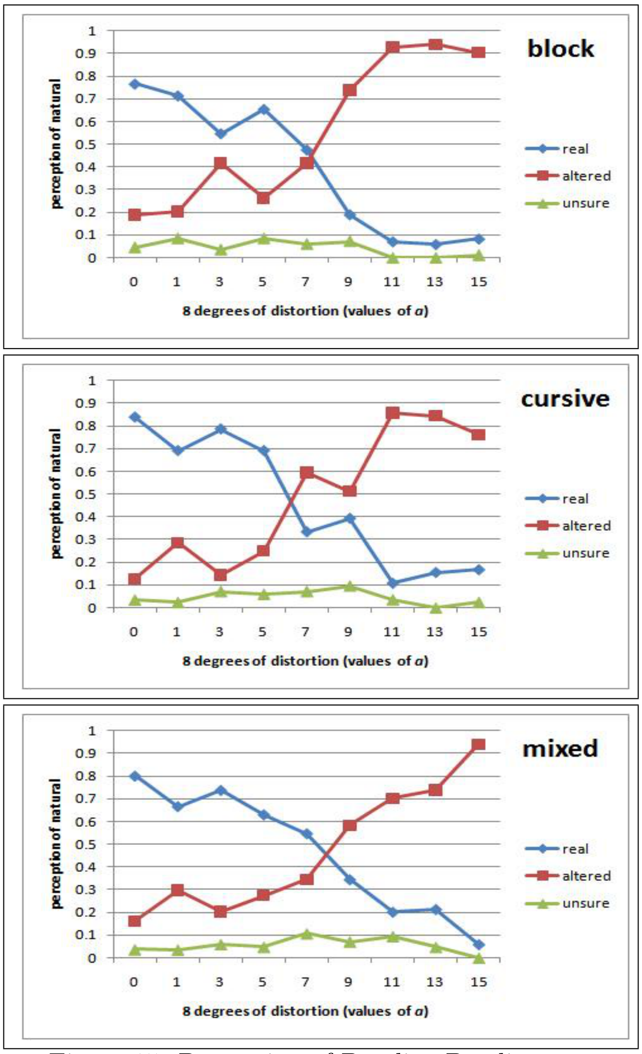

Figure 15: Perception of Baseline Bending.

For Shearing. The results for cursive and mixed handwriting are similar, with the intersections between a=0.31 and a=0.37. However, the intersection for block handwriting is much smaller, around a=0.23. This suggests that within a textline, variations in character angling is more easily discernable and perceived as unnatural with block lettering.

For Horizontal Scaling. The intersections of "real" and "altered" lines are at similar positions for all three writing styles.

For Vertical Scaling. The intersections for all three styles occur approximately at the same position. It is worth noticing that there is almost another intersection for cursive style. This may be noise due to the small number of subjects, or the fact that some of the cursive handwriting samples happened to be very tall, so that even small degrees of proportional distortions can be easily detected in an absolute height sense.

For Baseline Bending. The distortion strengths where the intersections of cursive, block and mixed handwriting occur increase slightly in that order.

Figure 16 shows the baselines of three writing styles, revealing how likely the subjects think the real handwriting is altered handwriting. As it can be seen, around 80% of real handwriting is thought to be real and around 15% is thought altered.

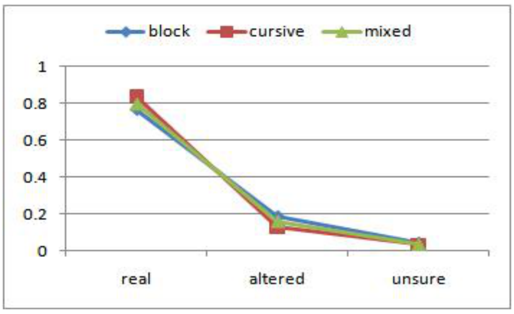

Figure 16: Likelihood that subjects think the handwriting is altered even when it is not.

Transformations	**50% Crossover Points**			l
	Block	Cursive	Mixed	
shearing	0.13-0.31	0.31-0.37	0.31-0.37	300
horizontal scaling	25-29	25-29	21-29	225
vertical scaling	0.26-0.35	0.29-0.38	0.23-0.35	150
baseline bending	5-9	5-7	7-9	230

Table 3: Approximately 50 crossover point by values of a.

Other Qualitative Observations:

- As shown in Figure 13, very light distortion is more likely to be thought altered with block writing (i.e., 0.3-0.4 of opinions) than cursive and mixed handwriting (0.1-0.25). The reason may be that expectations for uniformity of spacing between letters are much higher with block lettering, so minor distortion is more easily noticed. In addition, we notice that the likelihood that the distorted block handwriting is judged as "altered" drops down when a=37, i.e. the strongest distortion. This probably means that the distortion caused by horizontal scaling is not as obvious as other transformations so subjects' opinions toward horizontal scaled handwriting are unstable.
- Messier handwriting makes it harder to detect distortions.
- The time a subject takes to make a decision seems to be quickest for strongly distorted handwriting, average speed with little or no distortion, and slowest for weakly distorted handwriting.
- Generally baseline bending is the easiest distortion to detect, and horizontal scaling is the hardest.
- Different subjects are sensitive to different distortions.
- Subjects use various strategies to make their decisions. Besides their first reaction, one common technique was to compare two or more identical letters in a sentence, e.g., the two "w" in the sentence, "We know not what is good until we have lost it".

There were unexpected results from certain test subjects. For example, one subject thought 90% of the altered handwriting was real, believing all the messy handwriting real and all the neat handwriting unnatural. Another subject thought all of a certain writer's textlines were altered since they looked like a font of greeting cards. Since we did not tell the subjects how the computer algorithms worked, they imaged some distortion scenarios we did not apply. For example, some subjects thought we put pieces of handwriting from different textlines into one textline. In addition, since the subjects were learning the ways that the handwriting was distorted, their opinions at the beginning of the test could be different from their opinions after they saw more textlines including those strongly distorted.

Because these are preliminary results, we could hardly say which distortion strengths can produce good quality synthetic data. However, we can predict that the distortion strengths a little bit left to the intersection points are probably reasonable upper bounds for parameter settings. Here we speculatively summarize a series of optimal ranges for parameters of different transformations and writing styles in Table 4.

Table 4: Recommended ranges for parameter a and values of l for different transformations.

Transformations	**Recommended Parameter Ranges**			l
	Block	Cursive	Mixed	
shearing	0.07-0.22	0.16-0.31	0.16-0.31	300
horizontal scaling	15-25	17-25	15-23	225
vertical scaling	0.17-0.26	0.19-0.29	0.14-0.23	150
baseline bending	3-6	3-6	4-6	230

5. CONCLUSIONS AND FUTURE WORK

In this paper, we applied and extended the Varga and Bunke perturbation model to improve the quality of synthetic handwriting generated by computer. Parameter settings experiments were conducted by showing the human subjects a mixture of real and variably-altered English-language handwriting and asking them to judge whether the handwriting was natural or not. The results showed that distortion parameter settings' impact on perception of what is natural differ for different writing styles. This suggests style-specific parameter settings will be needed when generating synthetic training data.

In the future, experiments will be conducted across different languages (English, Chinese, and Arabic) and comparisons will be performed among languages. In the new experiments, besides parameter a, parameter l, i.e. the length of half-period cosine waves will also be varied. We will use random values of l to increase the variability of the training and testing set. Furthermore, synthetic handwriting using all four transformations together (full distortion) will also be examined. Finally, the time a subject spent on judging a textline might contain helpful information for fine-tuning parameter guidelines.

Since different parameter settings seemed to be necessary for different kinds of writing, it is natural to expect that this is also true for different languages. While it is always preferable to train a system using real labeled data, that is an expensive and time-consuming task. Therefore knowing how to create and consume high quality synthetic data can be valuable in building cost-effective handwriting recognition systems for new languages. There may also be some relevance in improving training systems for the related problem of writer identification, but it is a different problem from handwriting recognition.

ACKNOWLEDGMENTS

This work is supported by a DARPA IPTO grant administered by Raytheon BBN Technologies. This report has had a lot benefit of insightful and constructive criticism by Dr. Glenn Pearson from National Library of Medicine. I also want to thank Jin Chen, from Computer Science and Engineering Dept. of Lehigh University, for his helpful feedback to both of our research and this report.

REFERENCES

1. Baird, H., "State of the art of image degradation modeling," *Proc. of 4th International Workshop on Document Analysis Systems (DAS2000)* (2000).
2. Rowley, H., Goyal, M., and Bennett, J., "The effect of large training set size on online japanese kanji and english cursive recognizers," *Proc. of 8th International Workshop on Frontiers in Handwriting Recognition*, 36–40 (2002).
3. Velek, O. and Nakagawa, M., "The impact of large training set on the recognition rate of off-line japanese kanji character classifiers," *Proc. of 5th International Workshop on Document Analysis Systems (DAS2002)*, 106–109 (August 2002).
4. Cano, J., Perez-Cortes, J., Arlandis, J., and Llobet, R., "Training set expansion in handwriting character recognition," *Proc. of 9th SSPR/4th SPR*, 106–109 (2002).
5. Varga, T. and Bunke, H., "Off-line handwriting textline recognition using a mixture of natural and synthetic training data," *Proc. of 17th International Conf. on Pattern Recognition (ICPR'04)* **2**, 545–549 (August 2004).

6. Varga, T. and Bunke, H., "Effects of training set expansion in handwriting recognition using synthetic data," *11th Conf. of the International Graphonomics Society*, 200–203 (2003).
7. Varga, T. and Bunke, H., "Comparing natural and synthetic training data for off-line cursive handwriting recognition," *Proc. of 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR'04)*, 221–225 (August 2004).
8. Drucker, H., Schapire, R., and P.Simard, "Improving performance in neural networks using a boosting algorithm," *Advances in Neural Information Processing Systems 5 [NIPS Conference]*, 42–49, Morgan Kaufmann Publishers Inc. (1993).
9. Guyon, I., "Handwriting synthesis from handwritten glyphs," *Proc. of the Fifth International Workshop on Frontiers of Handwriting Recognition (IWFHR'96)* (1996).
10. Mao, J. and Mohiuddin, K., "Improving ocr performance using character degradation models and boosting algorithms," *Pattern Recognition Letters* **18**, 1415–1419 (1997).
11. Setlur, S. and Govindaraju, V., "Generating manifold samples from a handwritten word," *Pattern Recognitino Letters* **15**, 901–905 (1994).
12. Zheng, Y. and Doermann, D., "Handwriting matching and its application to handwriting synthesis," *Proc. of 8th International Conference on Document Analysis and Recognition (ICDAR'05)*, 861–865 (2005).
13. Mori, M., Suzuki, A., Shio, A., and Ohtsuka, S., "Generating new samples from handwriting numerals based on point correspondence," *7th International Workshop on Frontiers in Handwriting Recognition (IWFHR'00)* (2000).
14. Varga, T., Kilchhofer, D., and Bunke, H., "Template-based synthetic handwriting generation for the training of recognition systems," *Proc. of 12th Conference of the International Graphonomics Society* (2005).
15. Varga, T. and Bunke, H., "Generation of synthetic training data for an hmm-based handwriting recognition system," *Proc. of 7th International Conference on Document Analysis and Recognition (ICDAR'03)* (2003).
16. Marti, U.-V. and Bunke, H., "Using a statistical language model to improve the performance of an hmm-based cursive handwriting recognition system," *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)* , 65–90 (2001).
17. Poskanzer, J., Henderson, B., et al., *PAMSCALE User Manual of Netpbm* (2009). http://netpbm.sourceforge.net/doc/pamscale.html.
18. Farooq, F., Govindaraju, V., and Perrone, M., "Pre-processing methods for handwritten arabic documents," *Proc. of 8th International Conference on Document Analysis and Recognition (ICDAR'05)*, 267–271 (2005).
19. Bernard, T. M. and Manzanera, A., "Improved low complexity fully parallel thinning algorithm," *Proc. of 10th International Conference on Image Analysis and Processing*, 215–220 (1999).
20. Cai, J. and Liu, Z.-Q., "Off-line unconstrained handwritten word recognition," **14**, 259–280 (2000).