Table 2: 95% confidence intervals of the maximum and minimum correct rates at fixed values of one parameter.

`blur` (e)	min (%)	max (%)
e 0.5	[95.03, 96.64]	[95.97, 97.41]
e 1.0	[95.31, 96.87]	[96.67, 97.97]
e 1.5	[94.16, 95.91]	[97.54, 98.64]
e 2.0	[90.87, 93.07]	[97.45, 98.57]
e 2.5	[82.19, 85.18]	[96.44, 97.79]
`thrs` (t)	min (%)	max (%)
t 0.2	[89.97, 92.27]	[97.54, 98.64]
t 0.25	[90.93, 93.11]	[97.50, 98.61]
t 0.3	[89.44, 91.79]	[96.44, 97.79]
t 0.35	[87.73, 90.26]	[96.67, 97.97]
t 0.4	[82.19, 85.18]	[95.87, 97.33]
`sens` (s)	min (%)	max (%)
s 0.0	[87.98, 90.49]	[97.54, 98.64]
s 0.025	[85.10, 87.86]	[97.50, 98.61]
s 0.05	[85.41, 88.15]	[97.25, 98.42]
s 0.075	[83.34, 86.24]	[96.63, 97.94]
s 0.1	[82.19, 85.18]	[96.16, 97.56]

tory of the Department of Electrical Engineering, University of Washington, Seattle, WA, Fall 1993.

[5] G. Nagy, personal communication.

[6] S.V. Rice, J. Kanai, T.A. Nartker, The Third Annual Test of OCR Accuracy, in *Information Science Research Institute, 1994 Annual Research Report*, University of Nevada, Las Vegas, 1994, pp. 11–38.

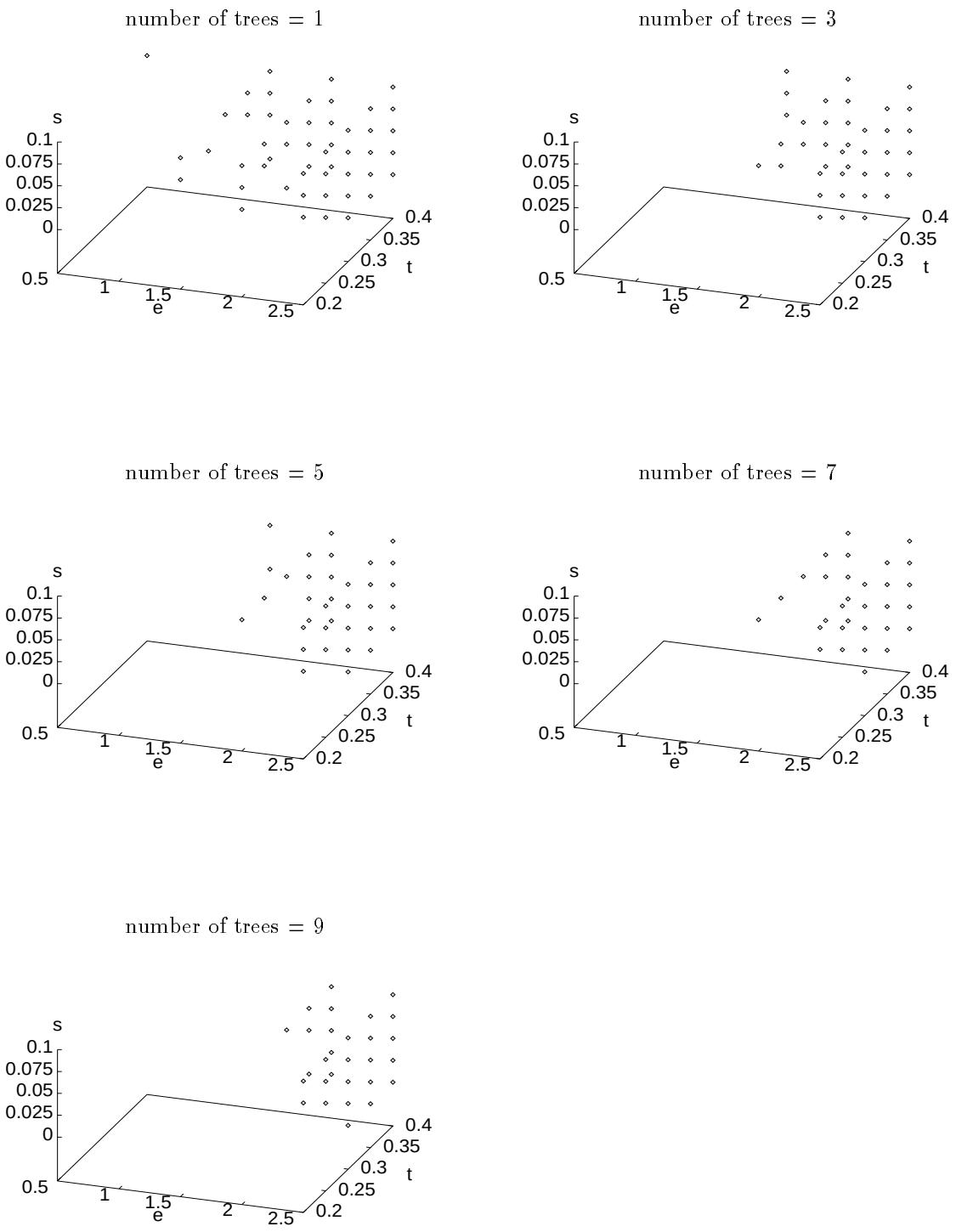

Figure 4: Parameter values associated with accuracies (% correct) below 96% as the complexity of the classifier grows.

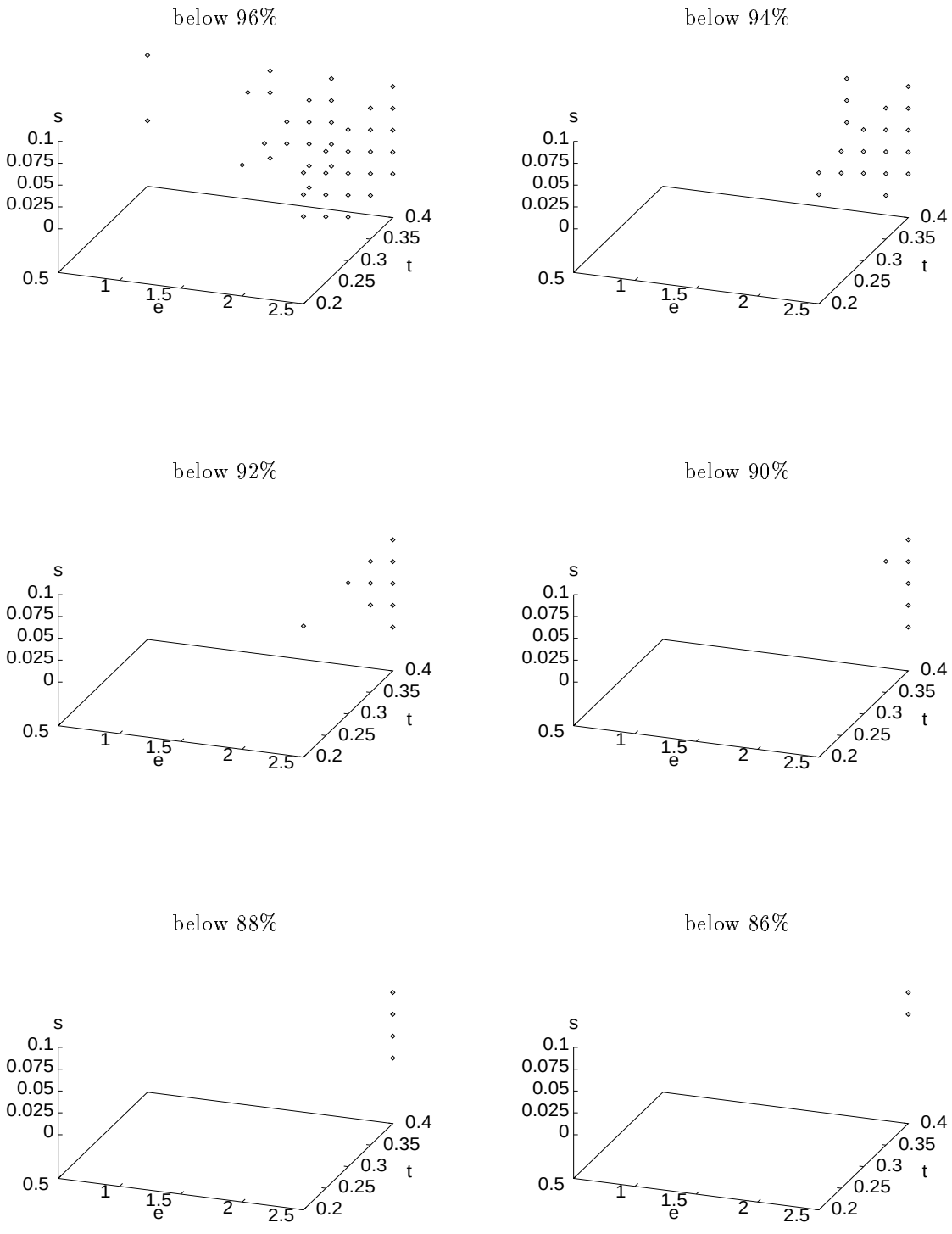

Figure 3: Parameter values associated with accuracies (% correct) below various thresholds.

sensitive to changes in `blur` and `thrs`, which become good predictors of its accuracy.

The effect of the parameters on accuracy varies among symbols. In other words, accuracy on some symbols is more sensitive to defects than others. For symbols

```
2469ACMkm
```

perfect (100%) accuracy is maintained throughout the entire range of defects we examined. The other symbols can be ordered as follows according to increasing difference between the highest and lowest correct rates over the region:

```
378JKPQXYbdghpqr#&$5BDEGHNRSV
WZacnuy@\+FLUesvx/^0OoTtz=%({
w>"?}f!~<j)]*[I:1l';i-'_|,.
```

Symbols towards the end of this queue are also those likely to be confused in the absence of context.

3.2 Domain of competence in the parameter space

The domain of competence of the classifier in the chosen parameter space can be shown as regions where the accuracy is above or below certain thresholds. Figure 3 shows the regions for accuracies below 96%, 94%, 92%, 90%, 88%, and 86% respectively. As noted before, the parameters associated with accuracies below each particular threshold tend to locate in contiguous regions in the space.

Similar maps can be made to compare the accuracies of different classifiers. For instance, the tree-based classifier we evaluate in this paper can be modified to use multiple trees and votings of their decisions. It has been observed that the accuracy improves as more trees are added to the classifier. To compare the accuracies, we map the values of the parameters associated with accuracies below 96% when different numbers of trees are used. Figure 4 shows such a map. It can be seen that the weak regions (where the values of the parameters are marked) shrink with the increase in the number of trees used.

4 Conclusions

We evaluated a character classifier using test data whose quality was systematically degraded. The data were synthesized with a document image defect model, with special attention to parameters controlling blurring, thresholding, and pixel sensor sensitivity.

We observed that blurring and thresholding, and to a lesser extent pixel sensor sensitivity, affect accuracy nearly continuously and monotonically, within measurement error. Thus good classifier performance is constrained to a contiguous region in parameter space. Blurring and thresholding had strong effects on classifier accuracy, and affected some symbols much more than others. Pixel sensor sensitivity plays an important role only when either of the other two parameters are at the margins of good performance.

In our evaluation we have assumed all symbols and all defects in the interesting range are equally probable. In real–world applications those probabilities vary, which need to be taken into account when accuracies are projected to realistic page images. The projection will also rely heavily on successful calibration of the model for real data.

References

[1] H.S. Baird, Document Image Defect Models, in H.S. Baird, H. Bunke, K. Yamamoto (Eds.), *Structured Document Image Analysis*, Springer–Verlag, 1992.

[2] T.K. Ho, *A Theory of Multiple Classifier Systems And Its Application to Visual Word Recognition*, Doctoral Dissertation, Department of Computer Science, SUNY at Buffalo, 1992.

[3] T.K. Ho, H.S. Baird, Asymptotic Accuracy of Two–Class Discrimination, *Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas, April 11–13, 1994, pp.275–288.

[4] English Document Database I & II CD–ROM Set, The Intelligent Systems Labora-

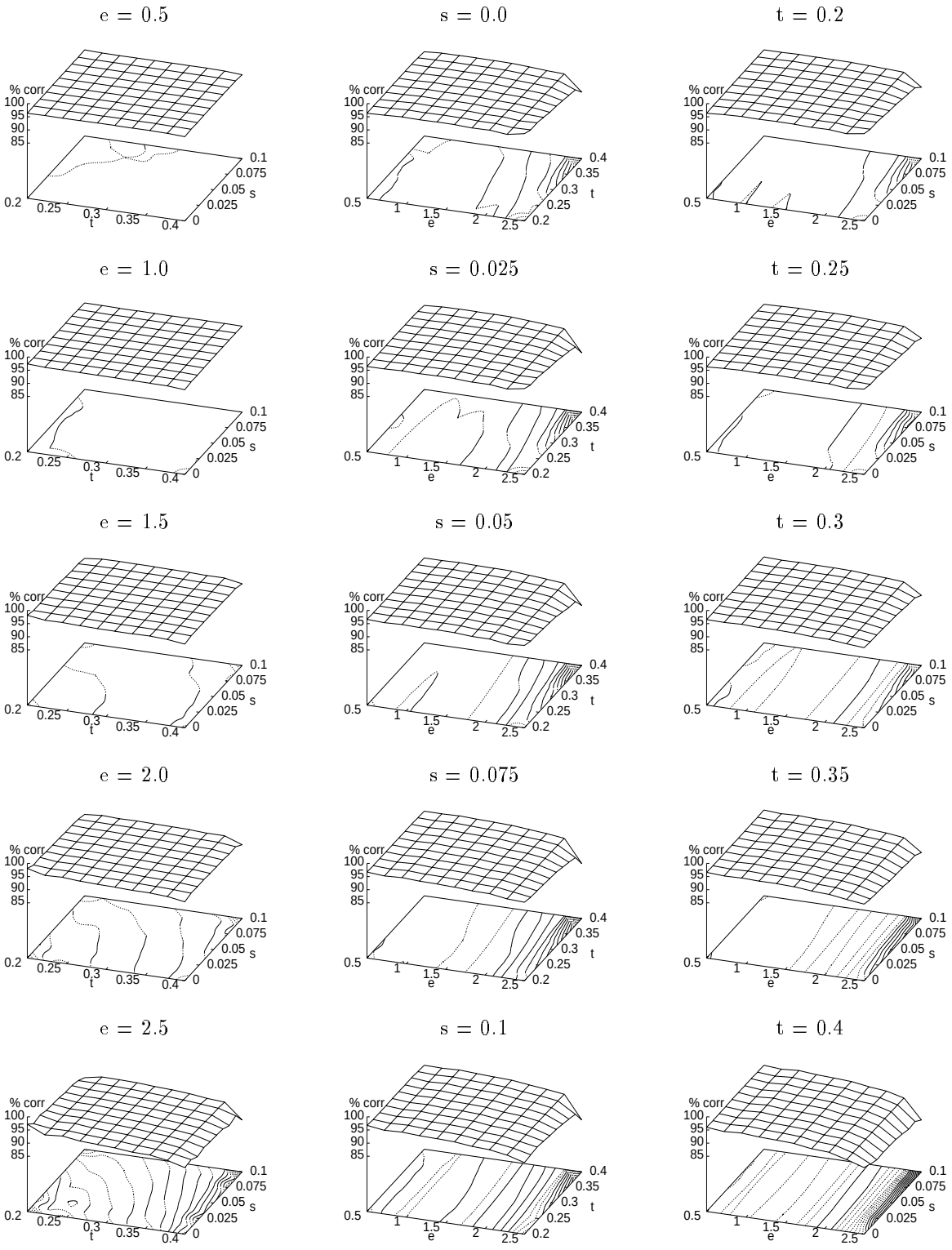

Figure 2: Accuracy plotted against parameters e (`blur`), s (`sens`), and t (`thrs`). The contours are at intervals of 0.5%.

Figure 1: Effects of blurring, thresholding, and pixel sensitivity. Images are created with `blur` varying from 0.0 to 3.6 by 0.4 (top row); `thrs` varying from 0.9 to 0.0 by -0.1 (middle row); and `sens` varying from 0.0 to 0.9 by 0.1 (bottom row).

```
!"#$%&'()*+,-./0123456789:;<=>?@
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
abcdefghijklmnopqrstuvwxyz{|}~
```

in the Adobe Times–Roman typeface. Thus a total of 6,250 samples were generated for each symbol. Half of these were used for training and the rest for testing.

We will illustrate our methodology on a classifier implemented as a binary decision tree where each internal node is associated with an oblique (i.e. non–axis–parallel) hyperplane [3]. The tree is derived automatically from the 293,750 ($25 \times 125 \times 94$) samples of training data. During testing, the 94 symbols are assumed to be equally probable.

3.1 Accuracy as a function of the parameters

Figure 2 shows the measured accuracy as a function of `blur`, `sens` and `thrs`. Except for a few cases where some images vanished, at each point the accuracy is averaged over 2,350 samples (25×94). Accuracy varies from 83.74% (occurred at {`blur`, `sens`, `thrs`} = {2.5,0.1,0.4}) to 98.17% (at {1.5,0.0,0.2}).

From Figure 2 we can see that over the region and at the coarse scale that we established, the accuracy appears to be continuous as a function of all three parameters. As a function of `blur` or `thrs`, it is monotonic (decreasing). Parameter `sens` has little effect except at extreme lower values of `blur` or `thrs`.

The fact that accuracy peaks at less than 100% is due to intrinsic ambiguities of shape such as case confusions and 'comma' *versus* 'single quote': these cannot be resolved in the absence of geometric and linguistic context. However, the consistently low accuracies at higher values of `blur` and `thrs` are certainly due to degradation in image quality.

The effect of each single parameter is shown in Table 2, which lists the 95% confidence intervals (C.I.s) of the maximum and minimum correct rates for each value of `blur`, `thrs` and `sens` over the variations of the other two parameters.

The C.I.'s for the minimum accuracies at each fixed value of the parameters shift over a wider range than those for the maximum accuracies. This suggests that each of the three parameters affects accuracy most when the other two parameters are at the margins of good performance. The disjoint C.I.'s of the minimum accuracies at the extreme values of each parameter suggest that their effects are statistically significant at the 0.05 significance level.

At higher values of `blur` ($\epsilon = 2.5$) or `thrs` ($t = 0.4$), the function fluctuates non–monotonically as `sens` changes, but the confidence intervals suggest that this is mostly due to measurement error. One conclusion from this experiment is that this particular classifier is

Table 1: Parameters of the image defect model.

Parameter	Effect	Units
randomized per–character ...		
`xresn`	horizontal output spatial resolution	pixels/inch
`yresn`	vertical output spatial resolution	pixels/inch
`size`	text size	points (1/72 inch)
`blur`	standard error of the Gaussian blurring kernel	pixels
`thrs`	binarization threshold	intensity
`skew`	rotation	degrees
`xscl`	horizontal scaling	dimensionless
`yscl`	vertical scaling	dimensionless
`xoff`	kerning, horizontal translation	pixels
`yoff`	height above baseline, vertical translation	ems
randomized per–char and per–pixel ...		
`sens`	variance of pixel sensor sensitivity	intensity
`jitt`	variance of jitter	pixels

most impossible to control in practice, and thus we allow them to vary randomly [5].

In our experience, aside from these, the key contributors to scanning and printing noise are the three parameters `blur` (standard error of the Gaussian blurring kernel), `thrs` (binarization threshold), and `sens` (pixel sensor sensitivity). These will be our primary concern. Figure 1 illustrates these effects.

The other parameters of the model are of secondary interest. The parameters `skew`, `xscl`, `yscl`, and `jitt` may have more or less effect depending on the features used in the classifier. Skew may have no effect with classifiers using rotation–invariant features. X and y scaling have little effect with classifiers which normalize character size. Jitter has effects similar to pixel sensitivity error. We defer studying these parameters.

Thus we designed our experiment as follows. We chose one fixed value for each of `size`, `xresn`, `yresn`, `skew`, `xscl`, `yscl` and `jitt`. We let `xoff` and `yoff` vary at random. We allow `blur`, `thrs`, and `sens` to vary in a controlled way over a range, and study the accuracy of the classifier as a function of these three.

Choosing values of the parameters

We fix the spatial sampling rate in both x and y directions (`xresn` and `yresn`) at 300 pixels per inch, and the text size `size` at 10 point. Skew is zero, scaling factors are 1.0, and jitter is disabled. We let `xoff` and `yoff` vary randomly uniformly within the range [-0.5, 0.5) pixels.

Parameters `blur`, `thrs` and `sens` vary within a range at fixed intervals: `blur` varies from 0.5 to 2.5 (inclusive) by a step of 0.5; `thrs` varies from 0.2 to 0.4 (inclusive) by 0.05; and `sens` varies from 0.0 to 0.1 (inclusive) by 0.025. These ranges were selected so that for most symbols the images neither vanish nor become shapeless black blobs. They are also consistent with the values used in the public–domain *Bell Labs BLidm0 Character Image Database* [4]. There are five values for each parameter, and thus 125 triples. We explore the behavior of the classifier over this lattice of 125 points in parameter space.

3 Classifier Evaluation

For each parameter point, we pseudorandomly generated 50 sample images for each of the 94 printable–ASCII symbols:

to one quality [2].

In the following sections, we will give a brief description of the model we used, and our choice of parameters for the evaluation. We will then describe our methodology along with results in evaluating a particular classifier.

2 Data Generation

The parameterized image defect model

To create test images we use a parameterized model of document image defects [1]. The model describes effects of the following factors: (1) the nominal text size of the output, (2) the output spatial sampling rate (both horizontally and vertically), (3) the point spread function (the standard error of a Gaussian blurring kernel), (4) the digitizing threshold, (5) the variation of sensitivity among the pixel sensors, (6) the variation of jitter among the pixels (*i.e.* discrepancies of the sensor centers from an ideal square grid, in units of output pixel), (7) rotation (skew angle), (8) stretching factors (both horizontally and vertically), and (9) translation offsets with respect to the pixel grid. Each parameter has a range of values that is determined by the physics of printing and imaging. Table 1 summaries the parameters corresponding to each factor.

Taking an "ideal" (a black and white image at high digitizing resolution, obtained from bitmaps or scalable outline descriptions purchased from typeface manufacturers) bitmap of a character as input, the generator creates one image when the value of each parameter is fixed and when `sens` and `jitt` are zero. The effects of pixel sensitivity and jitter are randomized per pixel, and the parameters `sens` and `jitt` specify the variance of a normal distribution around zero mean. Multiple images can be created when `sens` or `jitt` is positive. Reasonable lower and upper limits can be set for most parameters including `skew`, `thrs`, `size`, `xresn` and `yresn`. Constraints can also be put on certain parameter values. For instance, we can require `xresn` = `yresn` or `xresn` = 2 × `yresn`.

The accuracy function

Our goal is to map the accuracy of a given classifier in this 12–dimensional parameter space. More precisely, we want to locate the region or regions within which the classifier's accuracy is above a certain threshold. The parameters are real numbers and there are infinitely many points in parameter space (`xresn`, `yresn`, and `size` are normally, but not intrinsically, integer–valued). Of course we can hope to estimate accuracy only at a finite number of these points.

The design of search strategies within parameter space will be aided if accuracy, as a function of the parameters, exhibits continuity and monotonicity properties. Informally, we say that a function is continuous if a small change in a parameter value does not cause large fluctuations in accuracy. It is monotonic if accuracy peaks along the boundary of the parameter range and falls off steadily away from the peak. If there is a single peak not on the boundary, but the function still falls off away from the peak, we say the function is bi–monotonic. Many optimization search strategies require the function to be no more complicated than bi–monotonic. We will investigate whether or not our classifier, parameterized by our image defect model, exhibits these helpful properties.

Strictly speaking, to establish continuity or monotonicity we need measurements at all scales, but this is computationally infeasible. We have adopted the following approximations: (1) choose a few parameters that are of greatest interest; (2) set a fixed sampling rate, ignoring smaller details; and (3) select a region of interest and examine the function only within that region.

Choosing parameters of interest

Some parameters are well known to have marked effects on the recognition accuracy of almost all classifiers: these include text size and spatial sampling rate [6]. However, in many applications, their values are known in advance. We will assume that they have a fixed value.

Variations in `xoff` and `yoff` are due to random errors in spatial sampling which are al-

Evaluation of OCR Accuracy Using Synthetic Data

Tin Kam Ho and Henry S. Baird
AT&T Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974, USA

Abstract

We describe an experiment in which an OCR classifier is evaluated on synthetic character images created using a document image defect model. The use of synthetic data permits control of the quality of the test data and quantitative analysis of the effects of image quality on OCR accuracy. For a particular classifier, we show that good performance — accuracy above a given threshold — occurs within contiguous regions of the space defined by the range of key parameters of the defect model.

1 Introduction

We are interested in developing a systematic strategy of evaluating and improving OCR accuracy. OCR accuracies reported in the literature are often measured on essentially arbitrary data sets collected in an *ad hoc* manner. The type of image defects contained in such data sets cannot be quantitatively specified. Also, due to the high cost of collection and truthing, these data sets are usually too small to provide uniform coverage of defects. The resulting biases limit the projectability of accuracy estimates to unseen data.

It is difficult to talk precisely about image quality at present. For instance, in [6] image samples are ranked in quality according to the median character recognition accuracy of six OCR systems. Such a ranking often does not agree with human judgement, and will change as OCR technology evolves: but, it does highlight the importance of quality.

We approach the problem by distinguishing several types of primitive physical models for defects, describing each with a statistical model, allowing them to vary independently, and combining their effects to produce images. Methods for calibration (estimating the parameters) and validation of such models are being intensively researched.

In this paper we describe early exploration of a methodology for evaluating an OCR classifier using synthesized image data created with an image defect model. Evaluation on synthetic images has several advantages: defect parameters are known precisely for the test data; comprehensive and uniform coverage of the range of defects is achievable; the test can be automated; and the sample size is not limited by the costs of manual truthing.

We foresee numerous applications of a methodology for systematic evaluation of image quality. If we could map the weaknesses of a classifier, in terms of image defects, then perhaps we could improve it by further training guided by this knowledge. If we could determine that the image quality of a given problem is capable of being narrowly characterized, we could perhaps build or select the best classifier for it, and perhaps estimate the achievable accuracy immediately without further testing. If we could determine that a problem consists of a mixture of image qualities, then perhaps we could combine several classifiers, each matched