Application of the Optimal Class-Selective Rejection Rule to the Detection of Abnormalities in OCR Databases

Thien M. HA
University of Berne
Institut für Informatik und Angewandte Mathematik
Neubrückstr. 10, CH-3012 Berne, Switzerland
Phone: +41 / 31 / 631 86 99
Fax.: +41 / 31 / 631 39 65
E-Mail: haminh@iam.unibe.ch

May 22, 1996

Abstract

Building large Optical Character Recognition (OCR) databases is a time-consuming and tedious work. Moreover, the process is error-prone due to the difficulty in segmentation and the uncertainty in labelling. When the database is very large, say one million patterns, human errors due to fatigue and inattention become a critical factor. This report discusses one method to alleviate the burden caused by these problems. Specifically, the method allows an automatic detection of abnormalities, e.g. mislabelling, and thus may contribute to clean up a labelled database. The method is based on a recently proposed optimum class-selective rejection rule. As a test case, the method is applied to the NIST databases containing nearly 300'000 handwritten numerals.

CR Categories and Subject Descriptors: I.5.0 [Pattern Recognition]: General; I.5.1 [Pattern Recognition]: Models; I.5.2 [Pattern Recognition]: Design Methodology; I.5.m [Pattern Recognition]: Decision.
Key Words: error estimation, truthing error, foreign handwriting style, segmentation error, sloppy handwriting.

Contents

1 Introduction

It has been recognised since a long time that the successful design of a pattern recognition system strongly depends on the availability of a large, representative and correctly labelled training set of data [12, 10]. Unfortunately, building such a database is a very time-consuming, tedious, and error-prone task. This report discusses one method that could help alleviating the burden of the building process. The method allows an efficient detection of abnormal patterns resulted from the building process.

The detection is based on the comparison between two error estimation methods. The first is the standard error-count method whereas the second results from the application of a recently proposed optimum class-selective rejection rule [7, 9]. The latter estimation method does not need the knowledge of the label of each pattern and is thus independent of the labelling [8]. Since the latter method does not make use of the labelling, it is insensitive to an eventual mislabelling which affects the first estimation method. Therefore, a discrepancy between the two methods would indicate the presence of some abnormalities in the building process.

To test the method, we use the NIST databases containing a large number of handwritten numerals (Sec. 1.1). Section 2 reviews the optimum class-selective rejection rule. Section 3 describes the detection method and the next section proposes a nomenclature of detected abnormalities.

1.1 NIST Databases

Two databases, namely, SD3 and SD7, were provided by the American National Institute of Standards and Technology (NIST) in 1992 as parts of a conference to assess the state-of-the-art in isolated handwritten character recognition [14]. Twenty-nine groups from Europe and North America participated to compare the performance of their OCR systems. In total, 47 systems, both commercial and research, were presented. The databases contain isolated numerals (digits) as well as upper- and lower-case letters. In this report, we describe only experiments involving isolated numerals from SD3 and SD7, which contain 223124 and 58646 numerals, respectively.

Table 1 shows our partition of the databases into training, validation, and test sets. The training and validation sets as well as the first test set (Test1) are subsets of SD3. The second test set (Test2) is identical to SD7, which has been recognised as having a statistical distribution different from that of SD3 [14].

2 Optimum Class-Selective Rejection Rule

In statistical pattern recognition, the probability that a given sample or pattern $x \in X$, the pattern space, belongs to the i^{th} class, in a N-class problem, is provided by the *posterior* probability $P(i/x)$ through the Bayes formula:

Database	NIST-SD3			NIST-SD7
Partition	1-40000	40001-50000	50001-223124	1-58646
Size	40000	10000	173124	58646
Use	Training	Validation	Test1	Test2

Table 1: Partition of NIST-SD3 and NIST-SD7 databases.

$$P_i(x) \equiv P(i/x) = \frac{p(x/i) \cdot \pi_i}{p(x)}; i = 1, .., N \tag{1}$$

where $p(x/i)$ is the i^{th} class conditional probability density function (p.d.f.), π_i is the *a priori* probability of observing the i^{th} class, $\sum_{i=1}^{N} \pi_i = 1$, and[1]

$$p(x) = \sum_{j=1}^{N} p(x/j) \cdot \pi_j \tag{2}$$

is the absolute probability density function [3, 5]. It follows immediately that the *posterior* probabilities sum up to 1, i.e.,

$$\sum_{i=1}^{N} P_i(x) = 1 \tag{3}$$

A decision rule examines the *posterior* probabilities $P_i(x)$ and assigns to the input pattern x a number of classes. The way in which assignment is achieved defines the decision rule. See [8] for a review of various decision rules.

By definition the optimum class-selective rejection rule minimises the error rate for a given average number of classes [7]. The error rate is given by

$$e = \int_X risk(x)p(x)dx \tag{4}$$

where $risk(x)$ is the (conditional) probability of making a wrong decision, for a given x.

$$risk(x) = 1 - \sum_{i \in Selected\ Subset} P_i(x) = \sum_{i \in Rejected\ Subset} P_i(x) \tag{5}$$

The average number of classes, $\bar{n}$, is expressed by

$$\bar{n} = \int_X n(x)p(x)dx \tag{6}$$

[1]Without loss of generality, it will be assume that $p(x)$ is nonzero over the entire pattern space X, otherwise the region over which $p(x)$ is zero is first deleted.

where $n(x)$ is the number of classes assigned to pattern x. The choice of $\bar{n} = E_X[n(x)]$ is natural, and more importantly, it can be directly estimated from experiments by the sample mean

$$\frac{1}{N_s}\sum_{i=1}^{N_s} n_i \tag{7}$$

where n_i is the number of classes assigned to pattern x_i, and N_s is the total number of patterns involved in the experiment.

The optimum class-selective rejection rule assigns to pattern x all classes whose *posterior* probability is greater than a pre-specified threshold t. If there exist no such classes, the rule simply selects the (a) single best class [7, 9]. The domain of the pre-specified threshold is

$$0 \leq t \leq \frac{1}{2} \tag{8}$$

When $t = \frac{1}{2}$, the rule is equivalent to the Bayes rule, i.e., select the (a) single best class. When $t = 0$, the rule assigns to a pattern all classes whose *posterior* probability is non-zero. In between, the rule dynamically selects an appropriate number (between 1 and N) of best classes. It should be clear that the time complexity of this rule is $\mathcal{O}(N)$.

It turns out that it is possible to express the error rate directly as a function of the average number of classes via the Stieltjes integral [8]

$$e(t_{ope}) = -\int_{t=0}^{t_{ope}} t \cdot d\bar{n}(t) \tag{9}$$

where 'ope' stands for operating. (For an introduction to the Stieltjes integral, see [13].)

The marvelous feature of the above equation is that it allows the computation of the error rate at any level t_{ope} from $\bar{n}(t)$ solely and that the latter can be estimated from unlabelled patterns, by just counting the average number of selected classes, see Eq. (7). In other words, the error rate at any level can be estimated without knowing the true classes of the patterns. (Such a property is also shared by other rules [1, 2].) In particular, the Bayes error rate is given by

$$e_{Bayes} = e(t_{ope} = \frac{1}{2}) = -\int_{t=0}^{\frac{1}{2}} t \cdot d\bar{n}(t) \tag{10}$$

3 Detection of Abnormalities

The detection of abnormalities is based on the comparison of two estimations of the $e(\bar{n})$ curve using two different methods. Both methods utilise the optimum class-selective rejection rule of the last section and therefore need a recognition system to estimate the *posterior* probabilities $P_i(x)$ (Eq. (1)). The first method computes the error rate at each rejection level t by counting the number of patterns for which the 'true' label provided by NIST does not belong to the *Selected Subset* chosen by the recognition system. The second method first computes, for each rejection level t, the average cardinality of the *Selected Subset* and then numerically integrates the Stieltjes integral of Eq. (9) to obtain an estimation of the error rate.

3.1 Calibration of the Recognition System

The recognition system used in our study is a combination of two subsystems. Subsystem 1 estimates the *posterior* probabilities by first extracting a projection-based feature vector from the input pattern and then feeding it to a fully connected feed-forward multi-layer perceptron with architecture 49 : 60 : 10 (60 hidden nodes). Subsystem 2 estimates the *posterior* probabilities by first extracting a contour-based feature vector from the input pattern and then feeding it to a fully connected feed-forward multi-layer perceptron with architecture 104 : 60 : 10 (60 hidden nodes). Details about the two feature extraction methods can be found in [6]. Both neural networks are trained with the back-propagation algorithm [11] using the first 40000 numerals from SD3 and the next 10000 numerals are used to control the stopping of the training process.

The combined system is built via a two-stage combination scheme. Let $\hat{P}_i^{(1)}(x)$ and $\hat{P}_i^{(2)}(x)$ be the estimated *posterior* probabilities of class i by subsystems 1 and 2, respectively. In the first stage we construct two new classifiers, namely, C_{low} and C_{high}. (The reason for which they are named 'low' and 'high' will becomes apparent later.) The i^{th} output of C_{low} is the product of the i^{th} outputs of the two subsystems, i.e.,

$$\hat{P}_i^{(1)}(x) \cdot \hat{P}_i^{(2)}(x) \tag{11}$$

whereas the i^{th} output of C_{high} is the arithmetic average of the i^{th} outputs of the two subsystems, i.e.,

$$\frac{\hat{P}_i^{(1)}(x) + \hat{P}_i^{(2)}(x)}{2} \tag{12}$$

In general the outputs of C_{low} do not sum up to 1, which is the case for C_{high}.

In the second stage the final i^{th} output is computed by linear interpolation between C_{low} and C_{high} via a weighting factor $w \in [0, 1]$, followed by a normalisation so that the final outputs sum up to 1:

$$\hat{P}_i(x) = \frac{1}{F_{norm}}[(1-w) \cdot \hat{P}_i^{(1)}(x) \cdot \hat{P}_i^{(2)}(x) + w \cdot \frac{\hat{P}_i^{(1)}(x) + \hat{P}_i^{(2)}(x)}{2}]; i = 1, .., N. \tag{13}$$

Of course, the weighting factor must be experimentally determined. This is achieved by searching for w that equalises the error rates estimated by two independent methods. The first estimation method is the standard error-count procedure. That is

$$\hat{e}_{error-count}(w) = \frac{number\ \ of\ \ errors}{N_s} \tag{14}$$

where *number of errors* is counted using the recognition system given by Eq. (13), and N_s is the total number of patterns, or sample size, involved in the experiment.

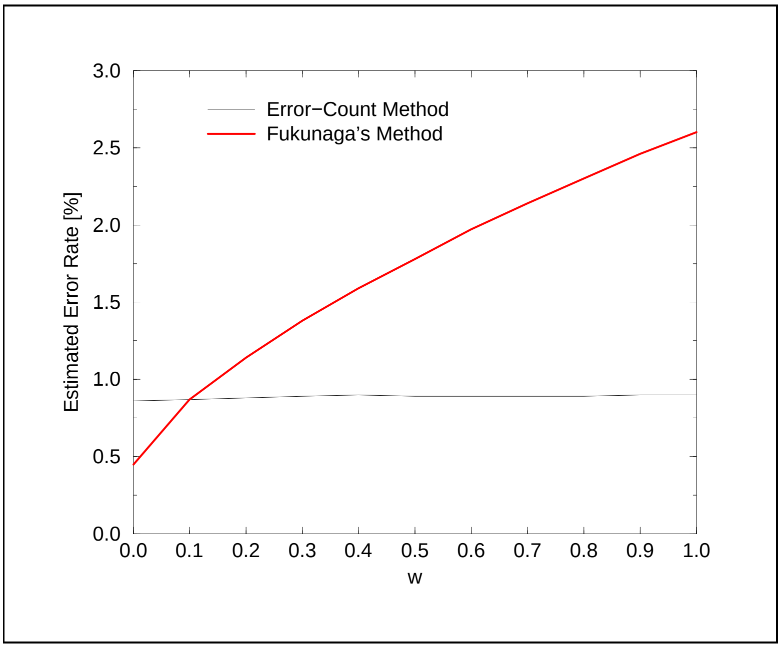

Figure 1: Error rate estimations using labelled (error–count method) and unlabelled (Fukunaga's method) patterns.

In our study we use the validation set from SD3 and thus $N_s = 10000$. The second estimation method was proposed by Fukunaga and Kessel [4]. This method estimates the error rate at zero rejection level by using only unlabelled patterns. The error estimate is given by

$$\hat{e}_{Fukunaga}(w) = \frac{1}{N_s} \sum_{j=1}^{N_s} [1 - \max_{i \in [1,..,N]} \hat{P}_i(x_j)] \tag{15}$$

In other words, the weighting factor w is chosen to be w^* so that

$$\hat{e}_{error-count}(w^*) = \hat{e}_{Fukunaga}(w^*) \tag{16}$$

Figure 1 shows the variations of $\hat{e}_{error-count}(w)$ and $\hat{e}_{Fukunaga}(w)$ for $w \in [0, 1]$. When $w = 0$, the method of Fukunaga under-estimates the error rate whereas it over-estimates the error rate for $w = 1$. This is the reason why the two new classifiers of the first stage were named C_{low} and C_{high}. It can be seen that $w^* = 0.1$, for which $\hat{e} \approx 0.87\%$, satisfies Eq. (16). This is the value that will be used in experiments throughout the report.

To check the validity of the calibration procedure, we use the calibrated system to estimate the error rates at zero rejection level on the data from Test1. Three estimation methods are used, namely, the error-count (Eq. (14)), Fukunaga's (Eq. (15)), and the new method via a discretised version of Eq. (10). Recall that only the first method makes use of the labels provides by NIST and the last two methods are solely based on the estimated *posterior* probabilities. The estimated error rates by these three methods are 0.535%, 0.559%, and 0.562%, respectively. It can be said that the last two methods provide estimated error rates remarkably close to the first one, despite the fact that they do not make use of the knowledge about the labels. The same experiment is repeated for the data from Test2, yielding the estimated error rates of 3.142%, 2.015%, and 2.023%, respectively. In this case the last two estimation methods differ significantly from the first one, although they do indicate that the error rate is higher than that of the validation set (0.87%). The most plausible explanation is that the data from Test2 (SD7) have a statistical distribution very different from those in the training and validation sets (SD3).

3.2 Detection by Comparison

The last section provides us with a calibrated recognition system. In this section we will use this system to compute the error–(average-number-of-classes), $e(\bar{n})$, curves by two independent methods, namely, the error–count method and the unlabelled method based on Eq. (9). At a first glance, the approach looks like the one for calibration of the last section. The main difference is that the calibration approach allows the error estimation only at zero rejection level whereas the current approach provides the error estimation at *all* levels.

For the error–count method, the threshold t is varied within its appropriate range according to Eq. (8). For each value of the threshold, the optimum class-selective rejection rule is applied to all test patterns yielding an error rate and an average number of classes. The produced pairs allow us to plot the $e(\bar{n})$ tradeoff curve. Of course, the estimation of the error rates is based on the pattern labels provided by NIST.

For the unlabelled method, the threshold t is varied within its appropriate range according to Eq. (8). For each value of the threshold, the optimum class-selective rejection rule is applied to all test patterns yielding an average number of classes. This results in an estimation of the $\bar{n}(t)$ curve. The error rates $e(t)$ are then estimated via a discretised version of the Stieltjes integral of Eq. (9). Therefore, for each value of t, a pair of numbers $(e, \bar{n})$ is obtained yielding an estimation of the $e(\bar{n})$ tradeoff curve. In this method, no labelling is needed and thus potential labelling errors in the truthing of the patterns would have no influence on the estimation.

Applying these two methods to the validation set of NIST gives the $e(\bar{n})$ curves on Fig. 2, where the error rate is presented on a logarithmic scale to emphasize small values.

The discrepancy between the two curves suggests that there must be something "wrong". More specifically, the two curves diverge for $\bar{n} > 2$: the curve obtained by the error–count method seems to be prematurely saturated. To get an insight of the problem, we attempt to detect the patterns that cause the discrepancy between the two curves. This is achieved by setting the threshold value to $t = 0.000251$ (for which $e = 0.02\%$ and $\bar{n} = 2$), applying the optimum class-selective rejection rule to all patterns of the validation set, and printing out those patterns that are considered

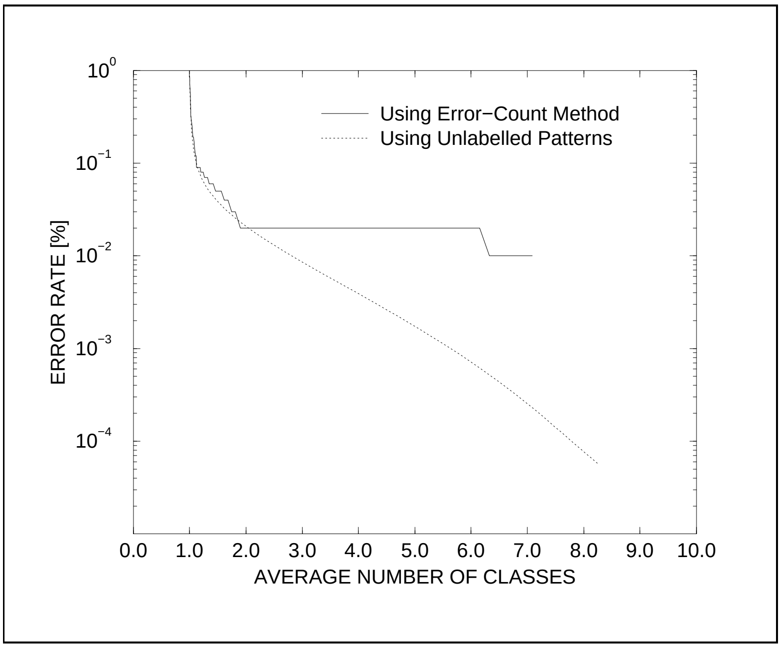

Figure 2: Estimation of the $e(\bar{n})$ curve by two independent methods. The data are from the validation set.

as errors according to the NIST labelling. Fig. 3 shows these patterns. Since NIST also provides the images from which these patterns were extracted (see Fig. 4), their inspection reveals that the labels are correct but the numerals had been wrongly segmented.

The detection method is efficient in that it requires human inspection only on a fraction (0.02% over 0.87%) of patterns considered as errors according to the labelling of NIST. Of course, it does not guarantee that all abnormalities are detected, but simply points out the most conspicuous ones. By increasing the threshold t, more abnormal patterns are detected; however, the additional detected patterns are less abnormal. Finally, it should be noted that some abnormal patterns may not be detected as errors, i.e., both the labelling and the recognition system make the same assignment error.

Since the validation set is a subset of NIST–SD3, it is suspected that similar abnormalities may be present in the remainder of the database. Therefore, we also apply the above detection method to the whole NIST–SD3. In order to be more confident in catching abnormal patterns, the threshold in slightly increased to $t = 0.0003$ so as to print out more patterns. Recall that when t increases, $\bar{n}(t)$ decreases while $e(t)$ increases. Fig. 5 shows the detected patterns; the corresponding file names are reported in Appendix A.1.

A similar study is also conducted for NIST-SD7. The comparison between the $e(\bar{n})$ curves obtained by the error–count and unlabelled methods is shown on Fig. 6. The saturation effect is found to start at $\bar{n} = 6$. The detected abnormalities are shown on Fig. 7; the corresponding file names are reported in Appendix A.2.

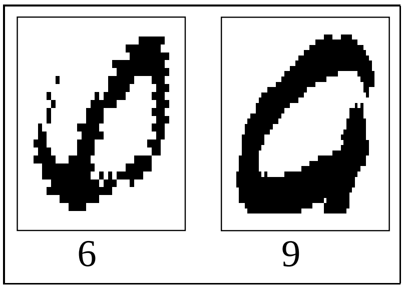

Figure 3: Abnormalities detected in the validation set. The digit under each pattern is the label provided by NIST.

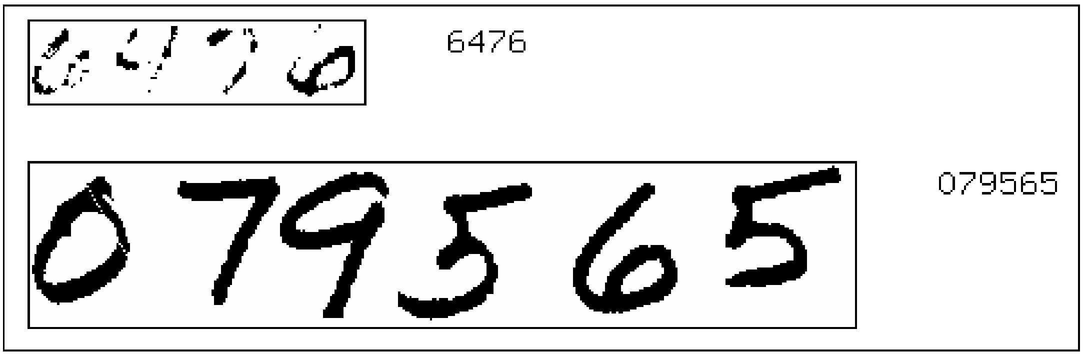

Figure 4: Original contexts of patterns in Fig. 3.

A simple inspection of these cases reveals that many different types of abnormality are present. For instance, apart from abnormalities due to segmentation, the pattern in the third row and second column of Fig. 5 is mislabelled. Another example is the foreign writing style (third row and last column of Fig. 5). A more detailed discussion on these aspects will be presented in the next section.

4 Classification of Abnormalities

The experiments of the last section show that there exist many different types of abnormality. In this section these abnormal patterns are grouped into four types, namely,

- Q : Quality,
- F : Foreign style,
- S : Segmentation, and
- M : Mislabelling.

The rationale behind this classification is the increasing degree of difficulty for the recogniser to deal with. Q-type patterns are the least serious since they can normally

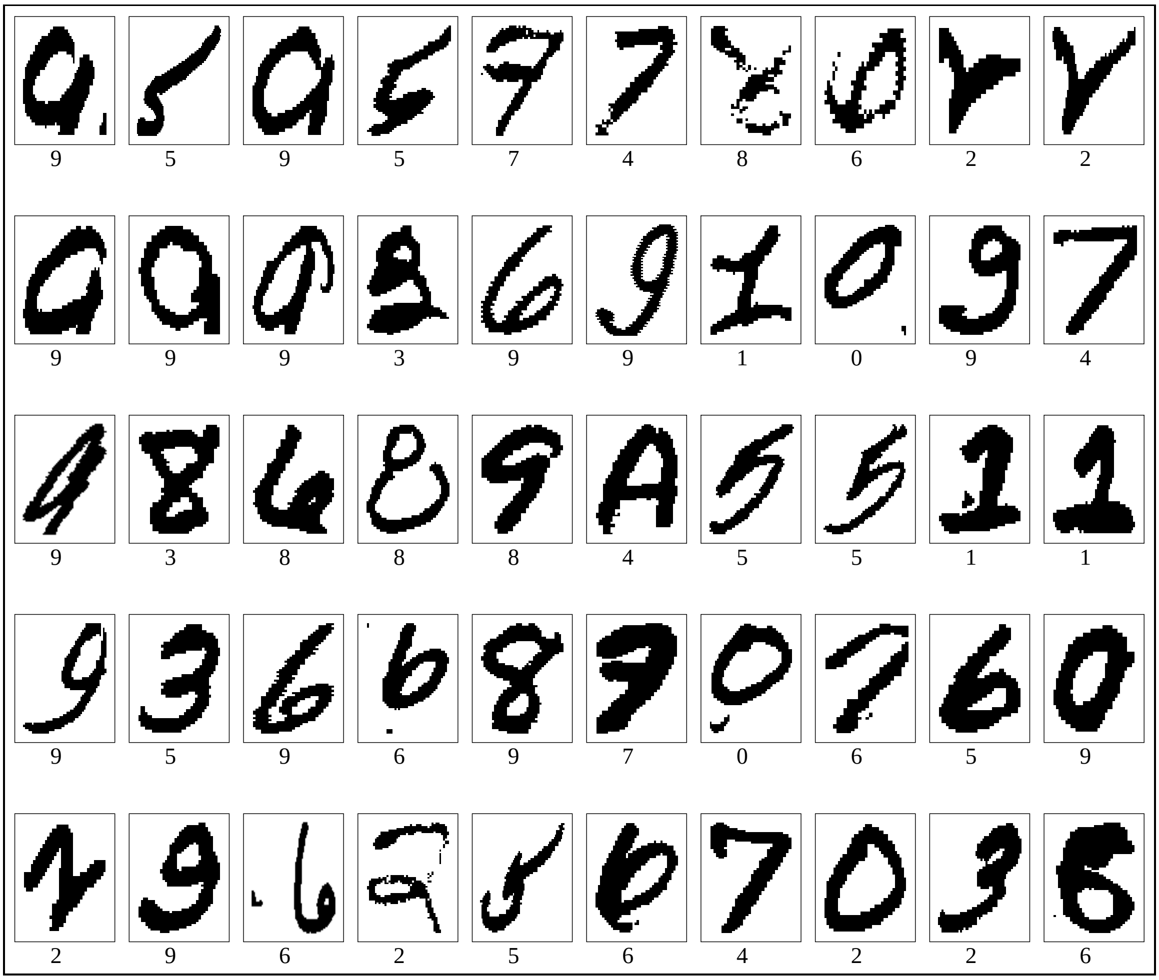

Figure 5: Abnormalities detected in NIST–SD3. The digit under each pattern is the label provided by NIST.

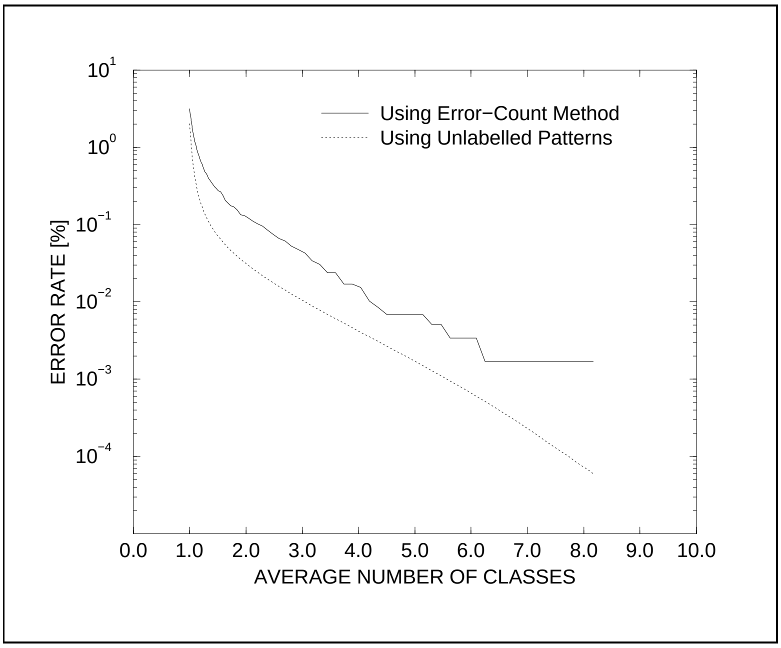

Figure 6: Estimation of the $e(\bar{n})$ curve by two independent methods. The data are from NIST–SD7.

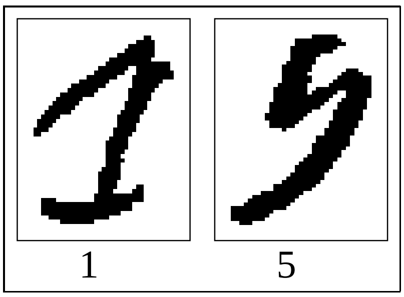

Figure 7: Abnormalities detected in NIST–SD7. The digit under each pattern is the label provided by NIST.

S	Q	S	Q	F	M	Q	S	Q	Q
S	S	S	Q	M	Q	F	S	Q	M
S	M	M	Q	M	F	Q	Q	F	F
Q	M	M	S	M	F	S	M	M	S
Q	Q	S	Q	Q	Q	M	M	M	Q

Figure 8: Classification of Fig. 5.

F	Q

Figure 9: Classification of Fig. 7.

be eliminated (or at least reduced) by increasing the size of the training database. F-type reveals the existence of non-representative patterns. S-type is more subtle in that it depends on how much information is discarded by bad segmentation. For instance, let us compare the pattern in row 3, column 1, and that in row 4, last column of Fig. 5. Both are from class '9', but the first one remains recognisable whereas the second does not. M-type patterns should simply be re-labelled.

Figures 8 and 9 shows the (manual) classification of Figs. 5 and 7, respectively, into the above four types.

5 Conclusion

We have proposed an efficient method for detecting abnormal patterns in a labelled database. The method is based on the analysis of the discrepancy between the error–(average-number-of-classes) curves estimated by two independent procedures. The application of the method to the NIST databases of handwritten numerals reveals the existence of different types of abnormality, for which a nomenclature is proposed. The method could be used as a tool to alleviate the burden of building large databases.

Acknowledgements: This work was partly supported by the Swiss National Science Foundation. The author would like to thank Matthias Zimmermann for numerous instructive discussions as well as proofreading, and Prof. H. Bunke for his constant encouragement.

A List of Abnormal Numerals

The syntax of each entry is

`Filename:Index`

A.1 NIST–SD3

```
sd3/data/hsf_0/f0087_24/d0087_24.mis:43
sd3/data/hsf_0/f0196_10/d0196_10.mis:48
sd3/data/hsf_0/f0235_17/d0235_17.mis:89
sd3/data/hsf_0/f0248_43/d0248_43.mis:67
sd3/data/hsf_0/f0253_40/d0253_40.mis:100
sd3/data/hsf_0/f0288_48/d0288_48.mis:14
sd3/data/hsf_0/f0340_05/d0340_05.mis:99
sd3/data/hsf_0/f0386_15/d0386_15.mis:89
sd3/data/hsf_0/f0394_00/d0394_00.mis:2
sd3/data/hsf_0/f0394_00/d0394_00.mis:97
sd3/data/hsf_0/f0407_39/d0407_39.mis:96
sd3/data/hsf_0/f0436_17/d0436_17.mis:29
sd3/data/hsf_0/f0498_06/d0498_06.mis:27
sd3/data/hsf_0/f0499_10/d0499_10.mis:85
sd3/data/hsf_1/f0592_11/d0592_11.mis:49
sd3/data/hsf_1/f0627_14/d0627_14.mis:63
sd3/data/hsf_1/f0636_12/d0636_12.mis:11
sd3/data/hsf_1/f0654_02/d0654_02.mis:66
sd3/data/hsf_1/f0683_02/d0683_02.mis:67
sd3/data/hsf_1/f0704_31/d0704_31.mis:22
sd3/data/hsf_1/f0719_05/d0719_05.mis:9
sd3/data/hsf_1/f0724_37/d0724_37.mis:98
sd3/data/hsf_1/f0724_37/d0724_37.mis:99
sd3/data/hsf_1/f0767_25/d0767_25.mis:6
sd3/data/hsf_1/f0828_34/d0828_34.mis:102
sd3/data/hsf_1/f0841_10/d0841_10.mis:43
sd3/data/hsf_1/f0930_07/d0930_07.mis:19
sd3/data/hsf_1/f0930_07/d0930_07.mis:42
sd3/data/hsf_1/f0933_38/d0933_38.mis:10
sd3/data/hsf_1/f0933_38/d0933_38.mis:20
sd3/data/hsf_1/f0944_27/d0944_27.mis:67
sd3/data/hsf_2/f1086_15/d1086_15.mis:38
sd3/data/hsf_2/f1190_46/d1190_46.mis:91
sd3/data/hsf_2/f1240_01/d1240_01.mis:4
sd3/data/hsf_2/f1268_12/d1268_12.mis:97
sd3/data/hsf_2/f1274_24/d1274_24.mis:85
```

```
sd3/data/hsf_2/f1281_28/d1281_28.mis:29
sd3/data/hsf_3/f1511_41/d1511_41.mis:44
sd3/data/hsf_3/f1672_35/d1672_35.mis:54
sd3/data/hsf_3/f1672_35/d1672_35.mis:77
sd3/data/hsf_3/f1750_06/d1750_06.mis:54
sd3/data/hsf_3/f1774_40/d1774_40.mis:29
sd3/data/hsf_3/f1817_18/d1817_18.mis:98
sd3/data/hsf_3/f1839_31/d1839_31.mis:92
sd3/data/hsf_3/f1867_44/d1867_44.mis:82
sd3/data/hsf_3/f1938_21/d1938_21.mis:96
sd3/data/hsf_3/f1944_40/d1944_40.mis:4
sd3/data/hsf_3/f2001_42/d2001_42.mis:106
sd3/data/hsf_3/f2015_08/d2015_08.mis:42
sd3/data/hsf_3/f2023_12/d2023_12.mis:53
```

A.2 NIST–SD7

```
sd7/test1/digit/d_001.mis:42
sd7/test1/digit/d_120.mis:100
```

References

[1] C.K. Chow, "On Optimum Recognition Error and Reject Tradeoff," *IEEE Transactions on Information Theory,* Vol. IT-16, No. 1, pp. 41-46, January 1970.

[2] P.A. Devijver, "Error and Reject Tradeoff for Nearest Neighbor Decision Rules," in G. Tacconi (Ed.) *Aspects of Signal Processing,*, Part 2, D. Reidel Publishing Company, Dordrecht-Holland, pp. 525-538, 1977.

[3] R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, 1973.

[4] K. Fukunaga and D.L. Kessel, "Application of Optimum Error-Reject Functions," *IEEE Transactions on Information Theory,* Vol. IT-18, pp. 814-817, November 1972.

[5] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, second edition, Academic Press, 1990.

[6] Thien M. Ha and H. Bunke, "Handwritten Numeral Recognition by Perturbation Method", *Proc. of the Fourth International Workshop on Frontiers of Handwriting Recognition,* Taipei, Taiwan, Dec. 7-9, 1994, pp. 97-106.

[7] Thien M. Ha, "An Optimum Decision Rule for Pattern Recognition," Technical Report IAM-95-009, Institute of Computer Science and Applied Mathematics, University of Berne, Switzerland, November 1995. (Anonymous ftp to **iamftp.unibe.ch**, Directory **pub/TechReports/1995/**)

[8] Thien M. Ha, "On Functional Relation between Recognition Error and Class-Selective Reject," Technical Report IAM-96-007, Institute of Computer Science and Applied Mathematics, University of Berne, Switzerland, March 1996. (Anonymous ftp to **iamftp.unibe.ch**, Directory **pub/TechReports/1996/**)

[9] Thien M. Ha, "An Optimum Class-Selective Rejection Rule for Pattern Recognition," *The 13th International Conference on Pattern Recognition,* Aug. 25-30, 1996, Vienna, Austria. To appear.

[10] J.J. Hull, "A Database for Handwritten Text Recognition Research," *IEEE Transaction on Pattern Analysis and Machine Intelligence,* Vol. 16, No. 5, pp. 550-554, May 1994.

[11] C.G.Y. Lau (Editor), *Neural Networks: Theoretical Foundations and Analysis,* IEEE Press, 1992.

[12] G. Nagy, "At the Frontiers of OCR," *Proceedings of the IEEE,* Special Issue on Optical Character Recognition, Vol. 80, No. 7, pp. 1093-1100, July 1992.

[13] S.M. Ross, *A First Course in Probability,* third edition, Macmillan Publishing Company, 1988.

[14] R.A. Wilkinson, J. Geist, S. Janet, P.J. Grother, C.J.C. Burges, R. Creecy, B. Hammond, J.J. Hull, N.W. Larsen, T.P. Vogl, and C.L. Wilson, *The First Census Optical Character Recognition Systems Conference,* The U.S. Bureau of Census and the National Institute of Standards and Technology, Technical Report #NISTIR 4912, Gaithersburg, MD, Aug. 1992.