Mathematical Symbol Recognition with Support Vector Machines

Malon, Christopher
Faculty of Mathematics, Kyushu University

Uchida, Seiichi
Faculty of Information Science and Electrical Engineering, Kyushu University

Suzuki, Masakazu
九州大学大学院数理学研究院

出版情報：MHF Preprint Series. 2007-7, 2007-02-06. 九州大学大学院数理学研究院
バージョン：
権利関係：

MHF Preprint Series

Kyushu University
21st Century COE Program
Development of Dynamic Mathematics with High Functionality

Mathematical symbol recognition with support vector machines

C. Malon, S. Uchida
M. Suzuki

MHF 2007-7

(Received February 6, 2007)

Faculty of Mathematics
Kyushu University
Fukuoka, JAPAN

Mathematical Symbol Recognition with Support Vector Machines

Christopher Malon [a,*], Seiichi Uchida [b], Masakazu Suzuki [a]

[a] *Faculty of Mathematics, Kyushu University*

Hakozaki 6–10–1, Higashi-ku, Fukuoka, 812–8581 Japan

[b] *Faculty of Information Science and Electrical Engineering, Kyushu University*

Motooka 744, Nishi-ku, Fukuoka, 819–0395 Japan

Abstract

Single–character recognition of mathematical symbols poses challenges from its two-dimensional pattern, the variety of similar symbols that must be recognized distinctly, the imbalance and paucity of training data available, and the impossibility of final verification through spell check. We investigate the use of support vector machines to improve the classification of InftyReader, a free system for the OCR of mathematical documents. First, we compare the performance of SVM kernels and feature definitions on pairs of letters that InftyReader usually confuses. Second, we describe a successful approach to multi–class classification with SVM, utilizing the ranking of alternatives within InftyReader's confusion clusters. The inclusion of our technique in InftyReader reduces its misrecognition rate by 41%.

Key words: Support vector machine; OCR; Mathematical document

* Corresponding author. Tel: +81 92 642 7047; Fax: +81 92 642 2789.
Email addresses: malon@math.kyushu-u.ac.jp (Christopher Malon),

1 Introduction

Mathematics is the universal language of scientific literature, but a computer may find it easier to read the human language in which surrounding text is written. The failure of conventional OCR systems to treat mathematics has several consequences:

- Readers of mathematical documents cannot automatically search for earlier occurences of a variable or operator, in tracing the notation and definitions used by a journal article.
- The appearance of mathematics on the same line as text often confounds OCR treatment of surrounding words.
- Equations can only be represented as graphics by semantic transformation systems, such as those converting digital documents into braille for accessibility by blind readers [13].

Mathematical OCR was investigated as early as 1968 [1]; a survey of its difficulties and previous approaches may be found in [2]. It differs markedly from typical text recognition because its single-character recognition phase must be followed by a *structural analysis* phase, in which symbol relationships such as superscripts, subscripts, fractions, and matrices must be recovered. The two-dimensional arrangement affects not only structural analysis but single-character recognition itself, because typical assumptions about bounding boxes and baselines are violated. Even in relatively simple equations such as

$$\phi|_{\mathbb{C}}(z) = \exp(zN_\phi)$$

uchida@is.kyushu-u.ac.jp (Seiichi Uchida), suzuki@math.kyushu-u.ac.jp (Masakazu Suzuki).

the subscript-positioned capital blackboard bold $\mathbb{C}$, whose base is nearly aligned with that of the vertical bar, might be mistaken for a lower-case letter.

Single-character OCR of mathematics also poses challenges that, if not unique, place it alongside the most difficult human languages to recognize. The recognition problem consists of about 1,000 classes, many with little existing ground truth data. Certain distinct letters, such as Latin v and Greek ν, are in close resemblance. Most unusually, we desire the distinction of *styles*.

In typical mathematical usage, different styles of the same letters will have completely different meanings. The problem is most severe not in engineering, but in pure mathematics. For example, within a single article in p-adic representation theory, the bold letter $\mathbf{G}$ often will represent a group over an algebraically closed field, the plain italic G will represent its rational points over a p-adic field k, and sans-serif G a reductive quotient over the residual field $\overline{k}$, with German $\mathfrak{g}$ used for a Lie algebra. Calligraphic $\mathcal{A}$ may represent a simplicial complex, and italic A a torus. (See, *e.g.*, [6].) An optical character recognizer that does not keep these letters distinct would be practically useless in this branch of algebra. However, within a single style, *fonts* (Computer Modern, Times, Helvetica, *etc.*) should not be distinguished, so that mathematical formulas can be compared between articles, regardless of the fonts the publisher has chosen.

In this paper, we present an experiment using support vector machines (SVM) to enhance single-character recognition of printed mathematics. OCR problems were considered very early in the development of SVM, with promising results. An experiment by Cortes and Vapnik [5] achieved 95.8% accuracy on handwritten digits in the US Postal Service database. More particularly,

in character recognition of human languages with hundreds of distinct characters, SVM have achieved promising results, for example, in handwritten Chinese (99.0%, [7]) and printed Ethiopian (91.5%, [11]). To our knowledge, the use of SVM in OCR of mathematics has not been investigated before.

This paper describes an experiment using SVM to improve multi-class classification by an existing OCR system. This OCR system is a purified version of the InftyReader, a freely available OCR engine for mathematics, described in [14]. First, we study the ability of various kinds of SVM, as binary classifiers, to distinguish pairs of letters that confuse InftyReader. Then, we show how the classifiers may be integrated with the system to improve its multi-class classification ability. Our results indicate that SVM is very suitable for mathematics symbol recognition.

2 Ground truth data

The InftyProject defined a set of 941 mathematical characters to be distinguished [15], and released several databases of ground truth, containing both single-character and structural recognition results. Because some mathematical symbols occur very rarely, it was necessary to choose between extracting each symbol from documents in their entirety, or seeking out samples of particularly rare characters to provide more uniform representation. Infty-CDB-3-B represents twenty articles from advanced mathematics journals at full length; it consists of a tenth of the samples of Infty-CDB-1, chosen by clustering techniques. The data of Infty-CDB-1 [15] is described in [16]. Infty-CDB-3-A [12] aims to represent rare characters by more samples; it includes not only journal articles, but font samples, and multiple scans of letters at different

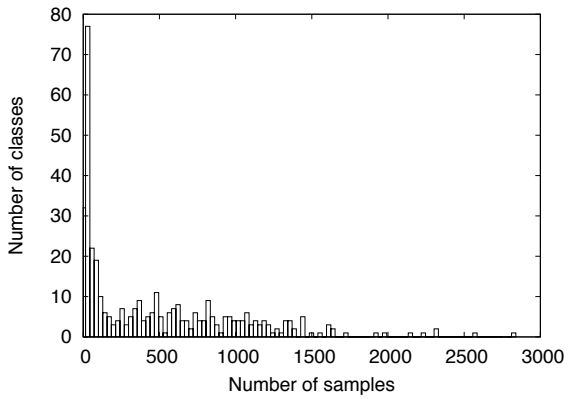


Fig. 1. Histogram of number of training samples, by class.

greyscale thresholds. We use Infty-CDB-3-A (188,752 characters, representing 384 symbol entities from 326 documents) for training, and Infty-CDB-3-B (70,637 characters, representing 275 symbol entities from 20 documents) for testing. No database includes samples of all 941 symbol entities defined by the Infty Project [16].

A sample of ground truth data for a symbol entity consists of a black and white bitmap image of that symbol in isolation, extracted from a scanned physical document. Bold letters are identified with their non-bold counterparts, because they are distinguished after single-character recognition, with the help of contextual information. German letters, which number too few, and touching and broken characters, are also excluded from our training and testing data.

In Table 1, we present representatives of the 384 symbol entities appearing in Infty-CDB-3-A. Figure 1 shows the number of training samples available for each of these classes.

3 Confusion Matrix

The engine of InftyReader typically makes use of contextual information; for this experiment, we distill it to ignore information about a character's size or surrounding characters. By running the purified InftyReader engine on the training data, we produce an integer-valued confusion matrix, with rows that count ground truth and columns that count recognition results. Every nonzero off-diagonal entry of this matrix represents a *confusing pair*, for which an SVM should be trained. There are 771 confusing pairs, counted as unordered pairs.

In the confusion matrix, each row represents Infty's recognition result and each column represents ground truth. The set of nonzero entries from each row of the confusion matrix represents a *confusion cluster*. The sizes of these clusters are indicated in Figure 2. As the figure shows, the confusion matrix is relatively sparse, and performing multi-class classification only on confusing alternatives, instead of all 384 symbols, significantly reduces complexity. Each cluster can be partially ordered by the likelihoods of each alternative, as indicated by the values of the corresponding matrix entries. This ordering will be utilized in our multi-class classification strategy later.

4 Pairwise classification with SVM

4.1 Features for SVM training

Some of our SVM are trained with directional mesh feature vectors, introduced by Kimura *et al* [10] for Japanese handwriting recognition. For a sin-

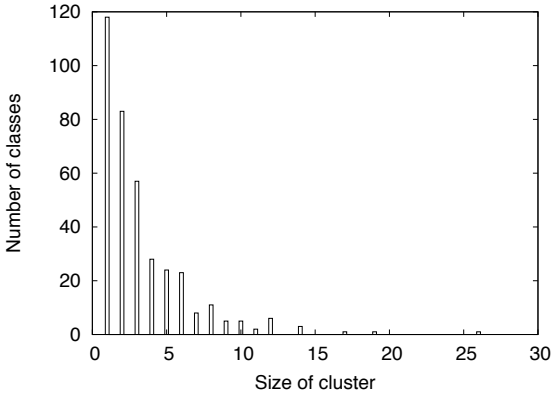


Fig. 2. Sizes of confusion clusters.

gle mesh, these feature vectors are constructed from directional histograms that measure the amount of contour pointing along four principal axes, in each position of the mesh. A single pixel contributes to the mesh position in which it lies, and possibly to several neighboring positions, as determined by mask functions over the bitmap. These mask functions sum to one everywhere in the bitmap. Experience in constructing the recognition engine for the InftyReader suggested that constantly square meshes capture insignificant information when a character is especially tall or short. Thus, our mesh data is divided into "tall," "square," and "short" blocks, representing the directional features from 3×5, 5×5, and 5×3 meshes, respectively. The set of blocks to be computed depends on the aspect ratio; one or two of the blocks will always be filled with zeros. The three blocks, together with the arctangent of the bounding box's aspect ratio, constitute our "direction" feature vectors, with dimension effectively (*i.e.*, excluding zero blocks) between 61 and 161.

In [17], Vapnik states the philosophy that, in contrast to classical approaches that work best with "strong features," "it is not important what kind of 'weak feature' one uses; it is more important to for 'smart' linear combinations." As

an extreme example of this philosophy, Cortes and Vapnik's original SVM study of the USPS handwritten digit database [5] utilizes (smoothed, centered, de-slanted) bitmap images as feature vectors. Bitmaps as feature vectors, sometimes transformed by principal component analysis, linear discriminant analysis, and nonlinear normalization, also have been the basis of more modern OCR experiments with SVM ([7], [4], and [11]).

To investigate whether the style-but-not-font distinction aspect of our recognition problem makes bitmap-based approaches less effective, or rather if directional features discard too much potentially useful information, we train another set of SVM with bitmap-like feature data. Because characters appear in bounding boxes of different aspect ratios, we cannot use raw bitmaps directly. Rather, we impose a 20 by 20 grid onto each bitmap, and measure the blackness in each grid position. Taking these measurements together with the arctangent of the aspect ratio, we obtain 401–dimensional "density" feature vectors.

4.2 Benchmark: A naive classifier

Ideally, we would compare performance of the SVM against the pure Infty recognizer itself. However, the pure Infty recognizer does not solve a binary classification problem like the SVM classifiers do. We can only say that the rate at which it picks a class A over a class B in binary selection should be greater than the rate at which it selects A out of all possible classes, and vice versa. These two bounds typically yield an interval too wide to be informative, so we implement a naive binary classifier as a more precise benchmark.

The naive classifier is constructed by recording the centroids of the sets of feature vectors representing instances of each symbol in the training data. We use the directional feature vectors for this construction. The naive classifier can perform either multi-class or binary classification; in any case, it assigns a test sample to the class with the closest centroid.

4.3 SVM training and performance

Altogether, we consider five forms of SVM constructions. On the directional features, we construct SVM with linear, Gaussian, and cubic polynomial kernels. On the density features, we construct SVM with linear and cubic polynomial kernels. These kernels have the forms:

$$K_{linear}(\vec{x}, \vec{y}) = \vec{x} \cdot \vec{y} \tag{1}$$

$$K_{Gaussian}(\vec{x}, \vec{y}) = e^{-\gamma \|\vec{x} - \vec{y}\|^2} \tag{2}$$

$$K_{cubic}(\vec{x}, \vec{y}) = (\gamma \vec{u} \cdot \vec{v} + 1)^3. \tag{3}$$

Support vector machines are trained to perform binary classification by solving the following optimization problem. Given training data with feature vectors $\vec{x_i}$ assigned to class $y_i \in \{-1, 1\}$ for $i = 1, \ldots, l$, the support vector machines solve

$$\begin{aligned} \min_{\vec{w}, b, \vec{\xi}} \quad & \frac{1}{2} K(\vec{w}, \vec{w}) + C \sum_{i=1}^{l} \xi_i \\ \text{subject to } & y_i (K(\vec{w}, \vec{x_i}) + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \end{aligned} \tag{4}$$

where $\vec{\xi}$ is an l–dimensional vector, and $\vec{w}$ is a vector in the same feature space as the $\vec{x_i}$ (see, *e.g.*, [8]). We use the LibSVM software [3] to train SVM classifiers.

The performance of a binary SVM classifier is evaluated according to a measure which we call the *min-accuracy*. This value is the smaller of its recognition rates for either class.

Before training an SVM, the soft margin C and any parameters appearing in the kernel K must be chosen in advance (here, γ). For the linear and Gaussian kernel experiments using directional features, we choose these parameters by five-fold cross validation. Each training document is assigned, in its entirety, at random to one of five sets. For each binary classification problem, the cross-validation accuracy for a choice of parameter values is computed by the leave-one-out method. Parameter choices are inspected within a grid in logarithmic space, and the grid is expanded until the accuracy stabilizes or begins decreasing at all its edges. The parameter choice producing the highest cross-validation accuracy is used once more to train the final SVM for the problem on the entire training set. This procedure cannot be performed on a binary classification problem if all the training data for either class is concentrated in a single one of the five sets; for the 17 (of 771) pairs where we have so little data, we do not construct a binary SVM.

In fact, the parameter choice was rarely important for the linear SVM; up to the hardest soft margin considered, accuracies typically remained the same, as one would expect if the data were linearly separable. A softer hard margin produces a 3% or greater improvement in min-accuracy on four pairs, and the constant choice $C = .01$ produces the best accuracies on training data overall. For the Gaussian kernel, as well, there is a parameter setting that yields cross-validation accuracies on each problem that are nearly as high as if the assignment is allowed to vary with the problem.

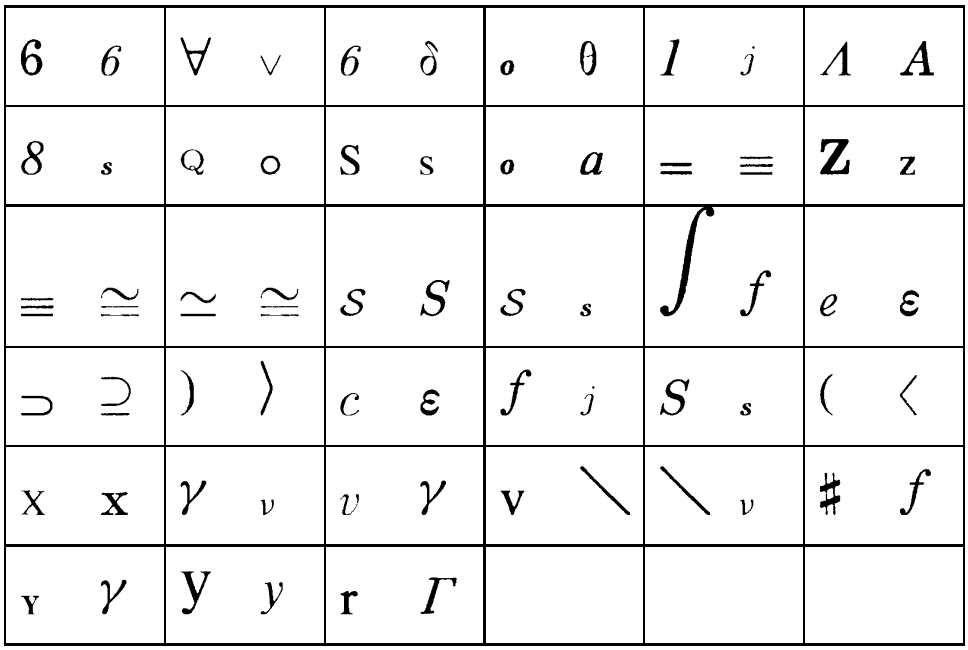


Fig. 3. Pairs on which the min-accuracy of linear SVM with directional features is at least 50% higher than that of the naive classifier.

To avoid the time expense of parameter selection, we use only a constant parameter selection to train the classifiers marked in Table 2 with an asterisk. These selections are made manually, based upon parameter search results for some particularly difficult pairs.

The binary classifiers are then evaluated on the testing data set. In Table 2, we compare their performance against each other and the naive classifier. This evaluation is only carried out for the 528 confusing pairs for which both classes have at least ten samples of testing data.

Table 2 shows that any SVM improves remarkably over the naive classifier, with no one kernel consistently better on directional features. Density features produce slightly worse results, particularly with the cubic polynomial kernel. The confusing pairs on which SVM achieves the greatest improvement in min-accuracy over the naive classifier are shown in Figure 3.

Because it is efficiently chosen, trained, and utilized, and as effective as the other classifiers, we will use the linear SVM on directional features as the basis for the analyses and multi-class experiments in the following sections. Remarkably, many distinctions are adequately learned by the linear SVM without much training data. Figure 4 plots the min-accuracy for each confusing pair against the number of samples in the smaller class of the pair.

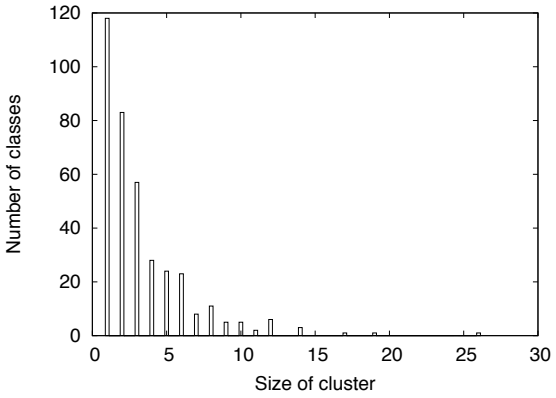


Fig. 4. Accuracy against the number of training samples.

5 Multi-class classification

By starting with a fast classifier, we reduce our multi-class classification problem from 941 classes to the size of the confusion cluster of an Infty recognition result, which can vary as shown in Figure 2. Popular methods of combining binary SVM to perform multi-class classification are reviewed in [9], including a method based on one-versus-all classifiers, and two methods based on one-versus-one classifiers (the max-wins and directed acyclic graph approaches). Each approach has well-known drawbacks, and none is suited to utilizing *a priori* information about the likelihood of alternatives, though the directed

acyclic graph method requires an order for the candidates to be chosen, whose implications are far from obvious.

One of these methods could be applied directly to a confusion cluster, but instead, we use a method that utilizes the ranking of alternatives in a confusion cluster, to make it likely that the most likely misrecognitions will be tested with an SVM.

For an Infty recognition result i, the confusion cluster $C(i)$ of misrecognition candidates is partially ordered by likelihood, as explained in Section 3. Let $C'(i)$ be the subset of alternatives $j \in C(i)$ for which a binary SVM comparing j and i was constructed. After the pure Infty engine recognizes a character as i, our method starts to apply the SVM for (j, i) for each $j \in C'(i)$, starting with the most likely j. When any j wins over i in the SVM classification, the testing is stopped, and j is reported as the classification. If no j wins, i is kept as the classification.

This method requires us only to train SVM on confusing pairs; other 1-versus-1 approaches would require us to train SVM on all pairs of letters that appear together in some confusion cluster. Of course, testing complexity is also linear in the number of letters in a cluster.

Without SVM, the pure Infty engine recognizes characters at a 96.10% accuracy on our testing data set. Using SVM by this method, the accuracy rises to 97.70%, marking a 41% reduction in misrecognitions.

When Infty makes the correct choice and our method does not, it always means that an SVM's decision was at fault. If neither Infty nor our method chooses correctly, three phenomena can explain the mistake. The SVM testing

the Infty's choice against the right alternative may have chosen the wrong result when it was reached (we count the cases where an SVM was not trained, because of insufficient data, as such a case). The confusion of Infty's guess for the correct answer might not have occurred in the training data, so that the right alternative was not represented in the confusion cluster; we call this situation an "unprecedented mistake." The final alternative is called "shadowing." On an instance of testing data for which Infty guesses i, and the correct answer is k, we say that an SVM is "shadowed" if some other alternative j occurs before k in the confusion cluster, and j defeats i, so that the i versus k classifier is never run.

Altogether, the classification on the 70,637 testing samples may be synopsized as follows:

- Infty right, output right: 67,100
- Infty wrong, output right: 1,912
- Infty right, output wrong: 784
- Infty wrong, output wrong, SVM wrong or not trained: 399
- Infty wrong, output wrong, unprecedented mistake: 280
- Infty wrong, output wrong, SVM shadowed: 162

If shadowing happened frequently, our multi-class strategy would be inappropriate, but this data shows that it happens quite rarely.

6 Style distinction

One novel aspect of our single-character recognition problem is the distinction of a letter in Roman, italic, calligraphic, and blackboard bold styles, regardless

of its font. The efficacy of SVM on this aspect of the problem is compared to that of other techniques in Table 3. In this table, only the top-ranked candidate selected by each method is compared to the correct answer.

The decrease in the number of confusing pairs means that the SVM can distinguish certain styles with 100% accuracy that pose confusion to other classifiers. The total number of style mistakes decreases from Infty to SVM by a greater margin than the misrecognition rate overall.

With occasional mistakes, the naive classifier typically can distinguish calligraphic and blackboard bold from other styles. Its main weakness is the distinction of italic characters. The linear SVM shows significant improvement in this regard. In Figure 5, we display three italic pairs that are markedly improved with SVM.

The only case where linear SVM performed remarkably worse than the naive classifier was in the distinction of lower case italic l from script lower case ℓ.

7 Summary

We have demonstrated the effectiveness of SVM on a large multi-class problem, with many similar symbols and many classes with little training data. The SVM managed to learn many binary classifications well for which there was a paucity of training data. Though all SVM kernels provided about the same performance on directional features, the linear classifier had superior performance on density features. Generally, SVM trained on directional features performed marginally better than SVM trained on density features. The SVM excels at distinguishing styles of characters, particularly italic and non-

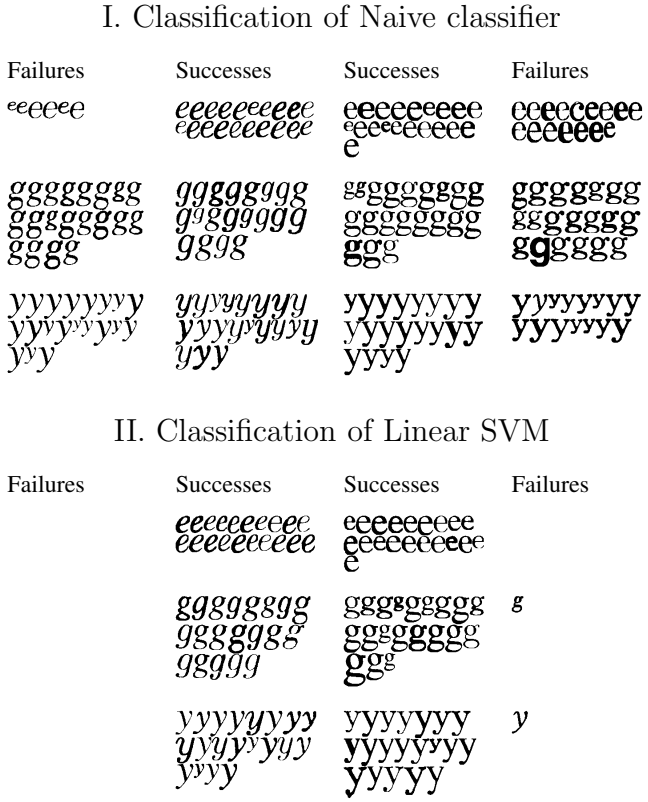


Fig. 5. Classification of the same letters in different styles.

italic variants, which are indistinguishable to simpler methods using the same sets of features.

We have integrated these SVM into the solution of a large multi-class problem, by testing only pairs of symbols mistaken by an existing OCR system. The complexity is low, and alternatives most likely to be confused are preferred by the algorithm. The single-character misrecognition rate of the OCR system has fallen by 41% since introducing SVM. We note that we have not omitted pairs often regarded as indistinguishable without size information (lower and upper case versions of C, O, P, S, U, V, X, and Z) in reporting our recognition rate.

Many of the mistakes that remain after the application of SVM represent char-

acters that are truly indistinguishable without contextual information (such as the character's size relative to surrounding characters), or that represent degraded character images. We will try to improve the use of contextual information in Infty, and develop better methods for the treatment of touching and broken characters, in future work.

References

[1] Anderson, R. *Syntax-directed recognition of hand-printed two-dimensional mathematics*. PhD thesis, Harvard University, 1968.

[2] Chan, K.-F., and Yeung, D.-Y. Mathematical expression recognition: a survey. *IJDAR 3*, 1 (2000), 3–15.

[3] Chang, C.-C., and Lin, C.-J. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/%7Ecjlin/libsvm`.

[4] Chang, F., Lin, C.-C., and Chen, C.-J. Applying a hybrid method to handwritten character recognition. In *ICPR '04: Proceedings of the 17th International Conference on Pattern Recognition* (Washington, DC, USA, 2004), vol. 2, IEEE Computer Society, pp. 529–532.

[5] Cortes, C., and Vapnik, V. Support-vector networks. *Mach. Learn. 20*, 3 (1995), 273–297.

[6] DeBacker, S. Stable distributions supported on the nilpotent cone for the group G_2. In *The Unity of Mathematics; In Honor of the Ninetieth Birthday of I.M. Gelfand*, Progress in Mathematics, vol. 244. Birkhäuser, Boston, 2006.

[7] Dong, J.-X., Krzyzak, A., and Suen, C. An improved handwritten chinese

character recognition system using support vector machine. *Pattern Recogn. Lett. 26*, 12 (2005), 1849–1856.

[8] Hsu, C.-W., Chang, C.-C., and Lin, C.-J. A practical guide to support vector classification. `http://www.csie.ntu.edu.tw/%7Ecjlin/papers/guide/guide.pdf`, July 2003.

[9] Hsu, C.-W., and Lin, C.-J. A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks 13* (2002), 415–425.

[10] Kimura, F., Wakabayashi, T., Tsuruoka, S., and Miyake, Y. Improvement of handwritten Japanese character recognition using weighted direction code histogram. *Pattern Recognition 30*, 8 (1997), 1329–1328.

[11] Meshesha, M., and Jawahar, C. Recognition of printed Amharic documents. In *ICDAR '05: Proceedings of the Eighth International Conference on Document Analysis and Recognition* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 784–788.

[12] Suzuki, M. Infty-CDB-3: a ground truthed database of words/formulae images, third distribution. `http://www.inftyproject.org/en/database.html`.

[13] Suzuki, M., Kanahori, T., Ohtake, N., and Yamaguchi, K. An integrated OCR software for mathematical documents and its output with accessibility. In *Computers helping people with special needs, 9th International Conference ICCHP 2004, Paris* (July 2004), Lecture Notes in Computer Science 3119, Springer, pp. 648–655.

[14] Suzuki, M., Tamari, F., Fukuda, R., Uchida, S., and Kanahori, T. Infty: an integrated OCR system for mathematical documents. In *DocEng '03: Proceedings of the 2003 ACM symposium on Document engineering* (New York, NY, USA, 2003), ACM Press, pp. 95–104.

[15] Suzuki, M., Uchida, S., and Nomura, A. A ground-truthed mathematical character and symbol image database. In *ICDAR '05: Proceedings of the Eighth International Conference on Document Analysis and Recognition (ICDAR'05)* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 675–679.

[16] Uchida, S., Nomura, A., and Suzuki, M. Quantitative analysis of mathematical documents. *International Journal on Document Analysis and Recognition 7*, 4 (2005), 211–218.

[17] Vapnik, V. *The nature of statistical learning theory.* Springer-Verlag New York, Inc., New York, NY, USA, 1995.

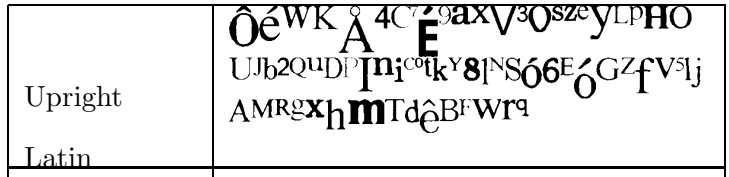
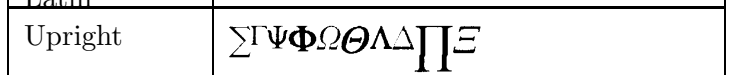
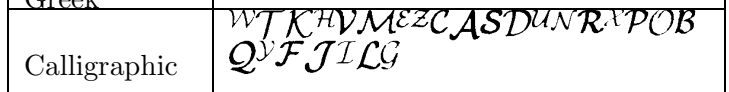
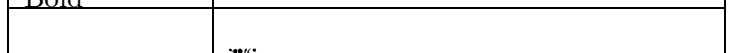
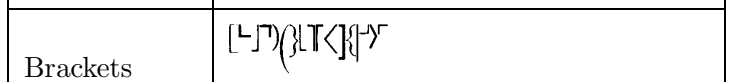
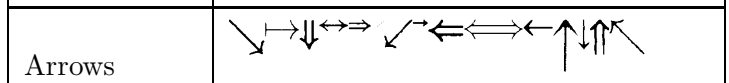
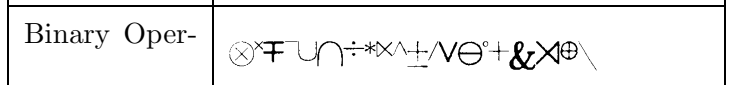
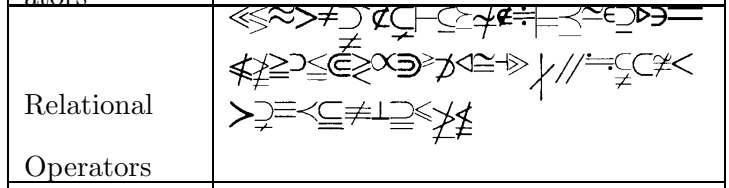
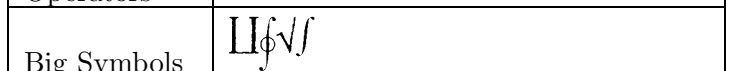
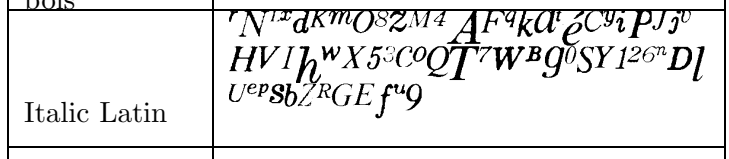
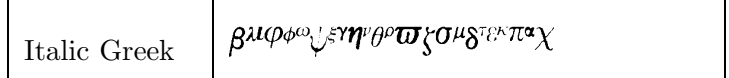
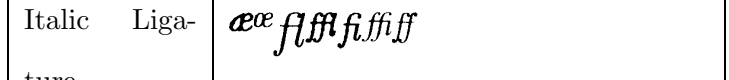
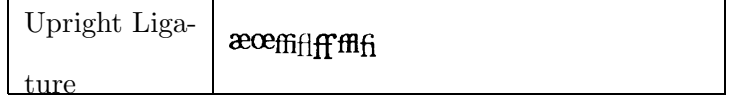
Upright Latin	
Upright Greek	
Calligraphic	
Blackboard Bold	
Punctuation	
Brackets	
Accents	
Arrows	
Binary Operators	
Relational Operators	
Big Symbols	
Other Symbols	
Italic Latin	
Italic Greek	
Italic Ligature	
Upright Ligature	

Table 1

Classes represented in Infty-CDB-3-A.

Percent of confusing pairs (total 528) attaining given accuracy

Accuracy	Naive Direction	SVM Direction Linear	SVM Direction Gaussian	SVM* Direction Cubic	SVM* Density Linear	SVM* Density Cubic
> 0	100.00%	100.00%	100.00%	100.00%	100.00%	96.21%
$> .5$	96.97%	99.05%	99.05%	99.05%	98.86%	94.51%
$> .6$	96.21%	98.48%	98.67%	98.48%	98.48%	94.32%
$> .7$	92.80%	98.30%	98.30%	98.30%	98.11%	92.99%
$> .8$	88.07%	97.35%	97.54%	97.35%	97.16%	91.10%
$> .9$	81.06%	95.83%	95.64%	95.64%	94.70%	88.07%
$> .95$	72.54%	92.99%	93.18%	92.61%	91.10%	83.33%
$> .97$	64.02%	90.34%	89.77%	89.96%	88.83%	78.22%
$> .99$	51.33%	84.28%	82.95%	84.28%	82.58%	70.83%
$> .995$	42.80%	78.22%	74.62%	77.84%	78.03%	66.29%
$> .999$	34.85%	69.13%	64.39%	69.89%	67.80%	51.14%

Table 2

SVM performance on confusing pairs.

	Naive	Infty	SVM
Total number of confused pairs	611	321	256
Confused pairs representing style mistakes	48	51	37
Total number of misrecognitions	8,466	2,753	1,625
Style recognition errors	871	219	116

Table 3

Style misrecognitions on testing data

List of MHF Preprint Series, Kyushu University

21st Century COE Program
Development of Dynamic Mathematics with High Functionality

MHF2005-1 Hideki KOSAKI
Matrix trace inequalities related to uncertainty principle

MHF2005-2 Masahisa TABATA
Discrepancy between theory and real computation on the stability of some finite element schemes

MHF2005-3 Yuko ARAKI & Sadanori KONISHI
Functional regression modeling via regularized basis expansions and model selection

MHF2005-4 Yuko ARAKI & Sadanori KONISHI
Functional discriminant analysis via regularized basis expansions

MHF2005-5 Kenji KAJIWARA, Tetsu MASUDA, Masatoshi NOUMI, Yasuhiro OHTA & Yasuhiko YAMADA
Point configurations, Cremona transformations and the elliptic difference Painlevé equations

MHF2005-6 Kenji KAJIWARA, Tetsu MASUDA, Masatoshi NOUMI, Yasuhiro OHTA & Yasuhiko YAMADA
Construction of hypergeometric solutions to the q - Painlevé equations

MHF2005-7 Hiroki MASUDA
Simple estimators for non-linear Markovian trend from sampled data:
I. ergodic cases

MHF2005-8 Hiroki MASUDA & Nakahiro YOSHIDA
Edgeworth expansion for a class of Ornstein-Uhlenbeck-based models

MHF2005-9 Masayuki UCHIDA
Approximate martingale estimating functions under small perturbations of dynamical systems

MHF2005-10 Ryo MATSUZAKI & Masayuki UCHIDA
One-step estimators for diffusion processes with small dispersion parameters from discrete observations

MHF2005-11 Junichi MATSUKUBO, Ryo MATSUZAKI & Masayuki UCHIDA
Estimation for a discretely observed small diffusion process with a linear drift

MHF2005-12 Masayuki UCHIDA & Nakahiro YOSHIDA
AIC for ergodic diffusion processes from discrete observations

MHF2005-13 Hiromichi GOTO & Kenji KAJIWARA
Generating function related to the Okamoto polynomials for the Painlevé IV equation

MHF2005-14 Masato KIMURA & Shin-ichi NAGATA
Precise asymptotic behaviour of the first eigenvalue of Sturm-Liouville problems with large drift

MHF2005-15 Daisuke TAGAMI & Masahisa TABATA
Numerical computations of a melting glass convection in the furnace

MHF2005-16 Raimundas VIDŪNAS
Normalized Leonard pairs and Askey-Wilson relations

MHF2005-17 Raimundas VIDŪNAS
Askey-Wilson relations and Leonard pairs

MHF2005-18 Kenji KAJIWARA & Atsushi MUKAIHIRA
Soliton solutions for the non-autonomous discrete-time Toda lattice equation

MHF2005-19 Yuu HARIYA
Construction of Gibbs measures for 1-dimensional continuum fields

MHF2005-20 Yuu HARIYA
Integration by parts formulae for the Wiener measure restricted to subsets in $\mathbb{R}^d$

MHF2005-21 Yuu HARIYA
A time-change approach to Kotani's extension of Yor's formula

MHF2005-22 Tadahisa FUNAKI, Yuu HARIYA & Mark YOR
Wiener integrals for centered powers of Bessel processes, I

MHF2005-23 Masahisa TABATA & Satoshi KAIZU
Finite element schemes for two-fluids flow problems

MHF2005-24 Ken-ichi MARUNO & Yasuhiro OHTA
Determinant form of dark soliton solutions of the discrete nonlinear Schrödinger equation

MHF2005-25 Alexander V. KITAEV & Raimundas VIDŪNAS
Quadratic transformations of the sixth Painlevé equation

MHF2005-26 Toru FUJII & Sadanori KONISHI
Nonlinear regression modeling via regularized wavelets and smoothing parameter selection

MHF2005-27 Shuichi INOKUCHI, Kazumasa HONDA, Hyen Yeal LEE, Tatsuro SATO, Yoshihiro MIZOGUCHI & Yasuo KAWAHARA
On reversible cellular automata with finite cell array

MHF2005-28 Toru KOMATSU
Cyclic cubic field with explicit Artin symbols

MHF2005-29 Mitsuhiro T. NAKAO, Kouji HASHIMOTO & Kaori NAGATOU
A computational approach to constructive a priori and a posteriori error estimates for finite element approximations of bi-harmonic problems

MHF2005-30 Kaori NAGATOU, Kouji HASHIMOTO & Mitsuhiro T. NAKAO
Numerical verification of stationary solutions for Navier-Stokes problems

MHF2005-31 Hidefumi KAWASAKI
A duality theorem for a three-phase partition problem

MHF2005-32 Hidefumi KAWASAKI
A duality theorem based on triangles separating three convex sets

MHF2005-33 Takeaki FUCHIKAMI & Hidefumi KAWASAKI
An explicit formula of the Shapley value for a cooperative game induced from the conjugate point

MHF2005-34 Hideki MURAKAWA
A regularization of a reaction-diffusion system approximation to the two-phase Stefan problem

MHF2006-1 Masahisa TABATA
Numerical simulation of Rayleigh-Taylor problems by an energy-stable finite element scheme

MHF2006-2 Ken-ichi MARUNO & G R W QUISPEL
Construction of integrals of higher-order mappings

MHF2006-3 Setsuo TANIGUCHI
On the Jacobi field approach to stochastic oscillatory integrals with quadratic phase function

MHF2006-4 Kouji HASHIMOTO, Kaori NAGATOU & Mitsuhiro T. NAKAO
A computational approach to constructive a priori error estimate for finite element approximations of bi-harmonic problems in nonconvex polygonal domains

MHF2006-5 Hidefumi KAWASAKI
A duality theory based on triangular cylinders separating three convex sets in R^n

MHF2006-6 Raimundas VIDŪNAS
Uniform convergence of hypergeometric series

MHF2006-7 Yuji KODAMA & Ken-ichi MARUNO
N-Soliton solutions to the DKP equation and Weyl group actions

MHF2006-8 Toru KOMATSU
Potentially generic polynomial

MHF2006-9 Toru KOMATSU
Generic sextic polynomial related to the subfield problem of a cubic polynomial

MHF2006-10 Shu TEZUKA & Anargyros PAPAGEORGIOU
Exact cubature for a class of functions of maximum effective dimension

MHF2006-11 Shu TEZUKA
On high-discrepancy sequences

MHF2006-12 Raimundas VIDŪNAS
Detecting persistent regimes in the North Atlantic Oscillation time series

MHF2006-13 Toru KOMATSU
Tamely Eisenstein field with prime power discriminant

MHF2006-14 Nalini JOSHI, Kenji KAJIWARA & Marta MAZZOCCO
Generating function associated with the Hankel determinant formula for the solutions of the Painlevé IV equation

MHF2006-15 Raimundas VIDŪNAS
Darboux evaluations of algebraic Gauss hypergeometric functions

MHF2006-16 Masato KIMURA & Isao WAKANO
New mathematical approach to the energy release rate in crack extension

MHF2006-17 Toru KOMATSU
Arithmetic of the splitting field of Alexander polynomial

MHF2006-18 Hiroki MASUDA
Likelihood estimation of stable Lévy processes from discrete data

MHF2006-19 Hiroshi KAWABI & Michael RÖCKNER
Essential self-adjointness of Dirichlet operators on a path space with Gibbs measures via an SPDE approach

MHF2006-20 Masahisa TABATA
Energy stable finite element schemes and their applications to two-fluid flow problems

MHF2006-21 Yuzuru INAHAMA & Hiroshi KAWABI
Asymptotic expansions for the Laplace approximations for Itô functionals of Brownian rough paths

MHF2006-22 Yoshiyuki KAGEI
Resolvent estimates for the linearized compressible Navier-Stokes equation in an infinite layer

MHF2006-23 Yoshiyuki KAGEI
Asymptotic behavior of the semigroup associated with the linearized compressible Navier-Stokes equation in an infinite layer

MHF2006-24 Akihiro MIKODA, Shuichi INOKUCHI, Yoshihiro MIZOGUCHI & Mitsuhiko FUJIO
The number of orbits of box-ball systems

MHF2006-25 Toru FUJII & Sadanori KONISHI
Multi-class logistic discrimination via wavelet-based functionalization and model selection criteria

MHF2006-26 Taro HAMAMOTO, Kenji KAJIWARA & Nicholas S. WITTE
Hypergeometric solutions to the q-Painlevé equation of type $(A_1 + A_1')^{(1)}$

MHF2006-27 Hiroshi KAWABI & Tomohiro MIYOKAWA
The Littlewood-Paley-Stein inequality for diffusion processes on general metric spaces

MHF2006-28 Hiroki MASUDA
Notes on estimating inverse-Gaussian and gamma subordinators under high-frequency sampling

MHF2006-29 Setsuo TANIGUCHI
The heat semigroup and kernel associated with certain non-commutative harmonic oscillators

MHF2006-30 Setsuo TANIGUCHI
Stochastic analysis and the KdV equation

MHF2006-31 Masato KIMURA, Hideki KOMURA, Masayasu MIMURA, Hidenori MIYOSHI, Takeshi TAKAISHI & Daishin UEYAMA
Quantitative study of adaptive mesh FEM with localization index of pattern

MHF2007-1 Taro HAMAMOTO & Kenji KAJIWARA
Hypergeometric solutions to the q-Painlevé equation of type $A_4^{(1)}$

MHF2007-2 Kouji HASHIMOTO, Kenta KOBAYASHI & Mitsuhiro T. NAKAO
Verified numerical computation of solutions for the stationary Navier-Stokes equation in nonconvex polygonal domains

MHF2007-3 Kenji KAJIWARA, Marta MAZZOCCO & Yasuhiro OHTA
A remark on the Hankel determinant formula for solutions of the Toda equation

MHF2007-4 Jun-ichi SATO & Hidefumi KAWASAKI
Discrete fixed point theorems and their application to Nash equilibrium

MHF2007-5 Mitsuhiro T. NAKAO & Kouji HASHIMOTO
Constructive error estimates of finite element approximations for non-coercive elliptic problems and its applications

MHF2007-6 Kouji HASHIMOTO
A preconditioned method for saddle point problems

MHF2007-7 Christopher MALON, Seiichi UCHIDA & Masakazu SUZUKI
Mathematical symbol recognition with support vector machines