Boosting Driven by Error Free Regions

Rainer Lindwurm, Jörg Rottland
Siemens Dematic AG
78459 Konstanz, Germany
{Rainer.Lindwurm, Joerg.Rottland}@siemens.com

Abstract

Multiple classifier systems improve the recognition performance of a discrimination task considerably, which makes them very attractive for pattern recognition products. Two aspects are eminently important: firstly, how can a powerful classifier ensemble be generated effectively and secondly, what classifier combination rule will produce the best collective result. This paper proposes a new boosting strategy, to generate a powerful classifier ensemble. The strategy trains a classifier ensemble by using sequentially selected learning sample subsets. The first subset is gained from the initial learning sample set. Each following subset is obtained from the previous steps subset by eliminating selected items. The selection criterion is a recognition quality limit, which divides the actual subset into error free and error containing result regions. The portion corresponding to the error containing region provides the basis for development of the next step classifier. The sampling subset is reduced iteratively until the discrimination with the last-trained classifier is almost errorless. A boost system, designed and developed in this way, shows excellent reclassification and a reduction of about 30 percent in the generalization error.

Keywords: Multiple classifier systems, boosting, character recognition

1 Introduction

Multiple classifier systems are not only an active topic of research, but also widely used in advanced recognition products. The deployment of such systems and methods ranges from the basic elementary classification units up to the combination of entire recognition devices [2]. The necessity for using multiple classifier systems has a long tradition that traces back to the very beginning of pattern recognition and artificial intelligence [3]. The main reasons for using multiple classifier systems were always the same: getting more recognition performance and reducing the needed computational power. In the past the second reason was often paramount: it was necessary to intelligently subdivide a complex classification task e.g. recognition of an alpha numeric class set [5], and then efficiently evaluate and combine the result afterwards, to tackle the hard time constraints. Nowadays, processor power has increased over several orders of magnitudes, bringing the enhancement of recognition performance more and more into focus. Sophisticated recognition algorithms and complex classifier structures running in parallel are no longer impossible and indeed they are absolutely necessary to fulfill the continually advancing requirements for recognition performance.

In the nineties, several methodologies were developed to create multiple classifier systems systematically. Two main representatives are *bagging and boosting*. The bagging strategy [1] is based on a random selection of elements from a given learning sample set, resulting in a specific sample subset. For each sample subset a separate classifier is calculated. Originally, in the recognition phase the results of all the subset classifiers are voted, to obtain the collective result. The boosting strategy [7] is based on an error driven sample subset creation. In a standard procedure [10] an initial classifier is adapted to the entire learning sample set. This initial classifier is used to identify all misclassified patterns. The misclassified patterns build the training basis for a secondary classifier. With this one, misclassified patterns are again identified, which leads to a further classifier. Each iteration generates an additional classifier element, and the sum of the elements constitutes the boost ensemble. Since the number of misclassified patterns is usually small, the main problem of this method lies obviously in the rapidly decreasing sample subset sizes. This often results in regularization problems in calculating the discriminant functions. To prevent this inconvenience, correctly recognized pattern must be reintroduced, at least to a certain extent. The strategies chosen and the specific sample set element weightings, give rise to various variants of this method.

Apart from how the classifier ensemble is generated, the choice of a method for combining the classifier results is of great importance. Algorithmic and adaptive approaches

Proceedings of the 9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004)
0-7695-2187-8/04 $20.00 © 2004 IEEE

are widely used and well examined [4] , [11] . The most well known algorithmic methods are the max-, min-, sum-, product rule or voting. The success of a selected combination method depends strongly on the specific system design of the multiple classifier system and the requirements of the present recognition application. Thus, for example achieving minimal error rates or maximal recognition rates requires different procedures, since these parameters cannot be optimized simultaneously.

In this paper a boosting strategy is introduced which does not incur the above mentioned regularization difficulties. This strategy also trains an initial classifier to the total learning sample set. Using this classifier an initial pattern classification of the total learning sample set takes place, which results in correctly and incorrectly recognized patterns. A recognition quality measure is calculated for each recognition result and this quality measure is used to generate the learning sample subset for the next step. If the quality measure of a sample set element is better than a given threshold, then the pattern is eliminated from the subset. If the quality measure of the sample element is worse than the quality measure threshold, then it is retained. The quality measure threshold is the quality value that separates the error free and error containing recognition result regions. Later in the recognition phase this threshold value is used together with the corresponding classifier. Thus the learning sample subset so obtained trains a further classifier. The procedure described above is repeated until the subsets size is so far reduced, that an errorless classification becomes possible. This last learning sample subset leads to a final classifier, which taken together with its predecessors makes up the boosting system.

In this boost system design, a hierarchical strategy is applied as classifier combination method. A pattern is classified by the initial classifier. If the recognition result indicates a recognition quality better than the corresponding threshold, then this classifiers result is accepted. Otherwise the next level classifier is invoked and the same test takes place. If the quality measure calculated for this pattern is again not acceptable for the second classifier, the next level processing is invoked. This is repeated until either the final classifier is reached and is forced to produce the final result or in other words, any intermediate classifier produced a quality measure better than the required threshold and its recognition result has already been accepted.

In section 2 our basic recognition method is introduced. In section 3 the classifier boost system design is described in more detail. In section 4 results are presented and finally in section 5 the conclusions are given.

2 Recognition Method

All the classifier elements of our multiple classifier systems are designed and developed as polynomial classifiers. Polynomial classifiers have a long tradition in our company [8] and in their widespread usage they have shown excellent performance and robustness over the years. Thus they can compete well with other methods like neural networks, multi reference systems and support vector machines [9]. Significant advantages of polynomial classifiers are their ability to be easily trained and flexibly modified and their unique solution due to the linear structure of the given discriminant function type.

In essence, the basic task of recognition is to find an underlying class structure, contained in a pattern set. The mapping of pattern elements to a specific class is based on features or measurements that first have to be calculated from the pattern. In the character recognition application, features may be gray level values of a raster image and the classes are a set of characters of a language, like numerals and upper or lower case letters. Since feature generation is usually a stochastic process, statistical methods are appropriate to solve the recognition task. It is well known, that the recognition task is solved [8], [6], if the corresponding a posteriori probability $p(y|x)$ is known. Here the target vector y indicates the class membership, and x represents the feature vector of the pattern under consideration. In the polynomial classifier context the a posteriori probability is approximated by a polynomial function.

$$p(y|x) \approx d(y|x) = A^T f(x) \tag{1}$$

d : Discriminant vector
A : Polynomial coefficient matrix, classifier
f : Polynomial extension of the feature vector

Hiding of the higher order degree feature values in a polynomial extension function has the enormous advantage that the basic equation remains linear and its solvability is preserved. So in principle, arbitrarily high order terms of feature variables can be introduced and processed with one and the same algorithm. The only limits arise from requirements imposed by the numerical calculations i.e. needed computer memory and computation time necessary for inverting the associated large dimensioned moment matrices.

In the training phase, the unknown polynomial coefficients have to be calculated with the help of a learning sample set. The optimization criterion used is the minimum mean square error S^2 of the sample set target vectors and the polynomial function approximation.

$$S^2 = E\{|y - d(x)|^2\} = \min_{d(x)} \quad (2)$$

$E\{\dots\}$: Expectation value

This criterion results in a matrix equation,

$$E\{f(x)f(x)^T\}A = E\{f(x)y^T\} \quad (3)$$

in which moment matrices appear e.g. $E\{f(x)f(x)^T\}$, that contain the statistical moments of the learning sample set. This equation can be solved by moment matrix inversion.

In the recognition phase the polynomial coefficients are known and the matrix product of the classifier with an unknown feature vector element generates a corresponding discriminant vector. The class decision is determined on the basis of the component with the maximum value of this discriminant vector.

3 Strategy and Boost System Design

The initial idea for the boosting strategy proposed here came from hints gathered from performance evaluation reports about executed recognition tests. In these reports the recognition results are distributed in a number of histogram bins according to their recognition quality measure rad. This measure is the distance of the recognition result d from the first choice target vector y_{max}: $rad = |d - y_{max}|$. The first choice target vector shows a value 1 for the maximum component of d and 0 for all the others. The rad-value has proved to be a powerful quality measure. Low values of rad indicate high quality recognition while high values of rad indicate a poor recognition. Recognition results with rad-values larger than 0.94 are counted as rejects. Additionally the distribution explicitly separates correctly from incorrectly recognized patterns. Thus the different distribution of both categories can be seen.

The distribution of the correct recognized patterns is concentrated on the low end of the rad-scale where better quality recognition results lie. Recognition errors lie at the higher end of the rad-scale where low quality recognition results are found. There is a range of overlap, where correct as well as incorrect recognition results occur. This is the most problematic range and it presents the fundamental challenge of the recognition task being considered. If both regions were well separated, zero error recognition would be possible. However, if the overlap range contains many patterns, a serious recognition problem exists. As can be seen in Figure 1, it happens, and this is not an exception, that in the low rad-range, no errors appear up to a certain threshold. If the underlying classifier is used up to this

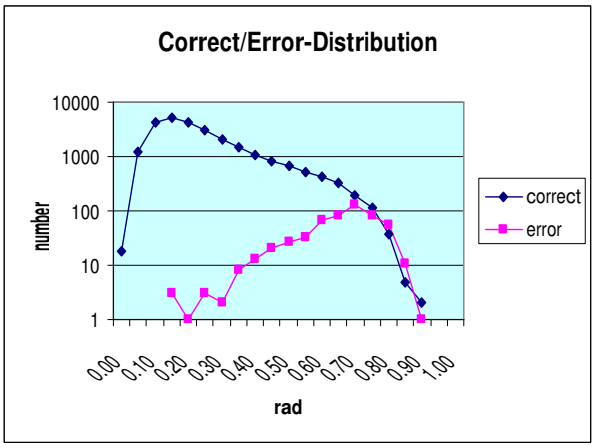

Figure 1. Correct-/Error-Distribution of learning sample set

threshold, no errors will occur. Reducing the initial learning sample set by these samples of high recognition quality leads to a smaller learning sample set and in so far to an easier recognition task.

A classifier trained with this reduced learning sample set has again a certain probability of recognizing a portion of samples without error, up a new threshold. This leads to further reductions of the learning sample set, from which a series of corresponding classifiers are trained, which should be operated just up to their threshold values, gained during the training phase. This procedure is repeated until the sample set size is so far reduced that a final classifier can be generated that recognizes the remaining samples almost entirely without errors (see Figure 2).

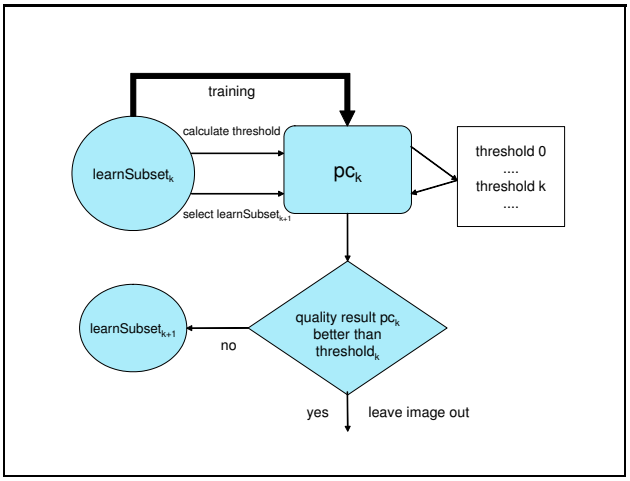

Figure 2. K-th step of training phase

pck: K-th polynomial classifier (pc) of boost system

In essence, this procedure works through the feature space in a specific, optimal way, and the hope is that this strategy will also be advantageous for any arbitrary test set. So during the training phase a series of classifiers are trained and associated thresholds are determined. These pairs, classifiers and thresholds, build the core of the new boost system.

During initial experiments, this strategy had to be slightly adjusted. It became evident that it was not always ensured that in each training level enough samples would remain to be selected. Especially, if any error occurred with a recognition quality measure equal to the best correctly recognized sample, the procedure entered an endless loop. So some times one or another error had to be accepted to ensure convergence of the strategy. This problem was not thus in the starting and end phases of the procedure, but in the middle range, where the real problem patterns are concentrated. It seemed that in the middle range there is an impenetrable labyrinth one had to fight through. Another modification was necessary, because it has to be ensured, that in each development level each class should be represented by some pattern. If this is not the case the procedure is terminated.

In the recognition phase, a pattern is sequentially classified. Each recognition of a pattern starts with the classification according to the initial classifier of the ensemble. From its recognition result the quality measure rad is calculated. If the quality measure is better than the initial classifiers threshold, the result is accepted and no other lower level classifier of the ensemble is invoked. Otherwise the pattern is handed over to the next lower level classifier, proceeding in the same way. This continues until the final classifier is reached from which a decision is then demanded (see Figure 3).

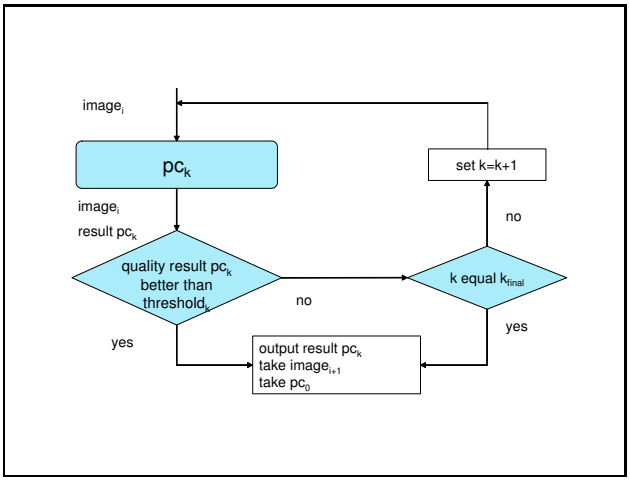

Figure 3. K-th step of recognition phase

4 Results

For evaluation of the proposed strategy, our learning and test sample set for hand printed numerals, EuroSiDe_Num_adap and EuroSiDe_Num_test, was used. The learning sample set consists of thirteen classes; each class consists of 2000 numerals, summing up 26000 numerals in total. The test set also has thirteen classes; each class consists of 500 numerals, summing up 6500 numerals in total. The thirteen classes arise because of the different writing styles in Europe for the numerals 0: 0 vs. $\emptyset$, 1: 1 vs. | and 7: 7 vs. 7. The classifiers in the boost system are trained to these thirteen classes but the recognition results and the used threshold values for the boost system are determined based on evaluations of the ten character classes only. Permutations between style classes belonging to the same character class are not counted as errors.

An initial classifier pc-B0 is trained with the total learning sample set. Applying the algorithm described in section 3 results in 10 additional classifiers pc-B1, . . . , pc-B10 with associated rad threshold values. These 11 classifiers together with the rad threshold values, build the boost system pc-Boost-V1 of the present application.

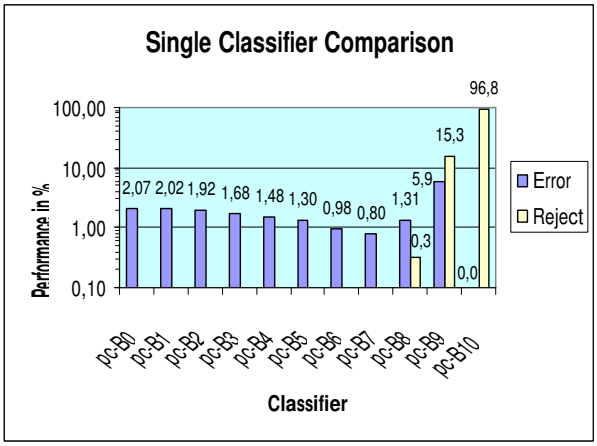

Figure 4. Single classifier performance on total learning sample set (reclassification)

In a first evaluation of the boost system, all individual classifiers are tested in isolation on the total learning and test sample set (see Figure 4 and Figure 5). All performance values shown in this paper are given in percent. Correct, error and reject rates sum up to 100 percent.

One interesting aspect of this test is that removing easily classified patterns i.e. those which have the best recognition quality measures, enhances the recognition performance. The minimum error rate could be achieved with the classifier pc-B7, which is trained with only about 25

percent of the initial training data, containing disproportionately many challenging patterns. This shows a certain similarity to the support vector philosophy, which focuses on the class border elements. Another interesting observation is that the performance characteristics of the individual classifiers are equivalent, due to the learning and test sample sets. This seems to be transferable to generalization. A further interesting point is that the procedure generates a single best classifier pc-B7, which hardly can be beaten by the total boost system.

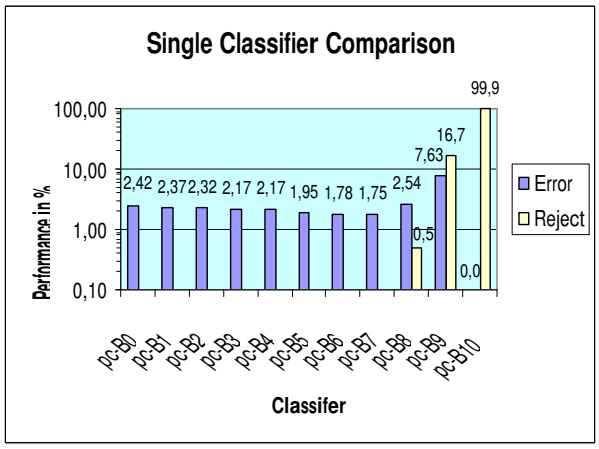

Figure 5. Single classifier performance on total test sample set (generalization)

A comparison of the initial classifier, the single best classifier and the straightforward boost system pc-Boost-V1 for the learning and test sample set is given in Tables 1 and 2.

Classifier	Correct	Error	Reject
pc-B0	97.92	2.07	0.01
pc-B7	99.20	0.80	0.00
pc-Boost-V1	99.95	0.05	0.00

Table 1. Performances on total learning sample set

The performance results show superb reclassification behavior of the boost system pc-Boost-V1. And even the low error rate is quite well maintained for the test sample set. But an unpleasant consequence is that in the generalization situation a considerable number of test samples seem to be unknown, and the boost system reacts with reject. The reject rate is nearly 16 percent and this is not acceptable for our applications. Therefore it was necessary to find a better final classifier for the boost system. To this end, various multi-classifier system methods were examined, like max rule and sum rule, but finally the single best classifier of the boost system itself, proved to be the best choice. So if the eleventh (last) classifier signals reject e.g. generated a rad-value larger than 0.94, the single best classifier is again invoked. The boost system constructed in this way is indicated as pc-Boost-Vopt. This resulted in the performance data, which still exhibits excellent reclassification but shows only a slightly further reduction in the generalization error (see Table 3).

Classifier	Correct	Error	Reject
pc-B0	97.58	2.42	0.00
pc-B7	98.18	1.75	0.07
pc-Boost-V1	83.91	0.11	15.98

Table 2. Performances on total test sample set

Classifier	Correct	Error	Reject
pc-B0	97.58	2.42	0.00
pc-B7	98.18	1.75	0.07
pc-Boost-V1	83.91	0.11	15.98
pc-Boost-Vopt	98.25	1.69	0.06

Table 3. Performances of different classifiers and boost systems on total test sample set

5 Conclusion

In this paper a boosting system strategy is presented, in which in a finite number of steps, a boost classifier system is generated. This system exhibits an excellent reclassification and this result is quite satisfying. With respect to generalization, the error rate in comparison to the initial classifier could be reduced by about 30 percent. Among all the classifier structures in use in our address reading products up to now, this is the best result we have ever attained, for this learning and test sample set. Surely, a larger generalization effect would have been welcome, considering the enormous additional effort spent on training and processing such a boost system. This matter remains to be explored by further research. The new strategy also produces a single best classifier in which, to a certain extent, the boosting effect is already incorporated, and this classifier is almost as good as the whole system. This offers the opportunity to run the single best classifier alone, if time or memory constraints are stiff. Intelligent classifier selection can certainly improve the found performances reached so far and this will be a direction for further activities.

References

[1] L. Breiman. Bagging Predictors. *Machine Learning Journal*, 24(2):123–140, 1996.

[2] A. Dengel, R. Hoch, F. Hönes, T. Jäger, M. Malburg, and A. Weigel. Techniques for Improving OCR-Results. In H. Bunke and P.S.P. Wang, editors, *Handbook of Character Recognition and Document Image Analysis*. World Scientific, Singapore New Jersey London Hong Kong, 1997.

[3] J. Franke. Zur Entwicklung hierarchischer Klassifikatoren aus der entscheidungstheoretischen Konzeption. In *5th Symp. Of DAGM, German Association on Pattern Recognition*. Springer-Verlag, Berlin Heidelberg New York Tokyo, 1983.

[4] J. Kittler. Improving Recognition Rates by Classifier Combination: A theoretical Framework. In A.C. Downton and S. Impedovo, editors, *Progress in Handwriting Recognition*. World Scientific, Singapore New Jersey London Hong Kong, 1996.

[5] R. Lindwurm, T. Breuer, and K. Kreuzer. Multi Expert System for Handprint Recognition. In A.C. Downton and S. Impedovo, editors, *Progress in Handwriting Recognition*. World Scientific, Singapore New Jersey London Hong Kong, 1996.

[6] T. Poggio and F. Girosi. A Theory of Networks for Approximation and Learning. *MIT A.I. Memo No. 1140*, MIT C.BV.I.P. Paper No 31:1–85, 1989.

[7] R.E. Schapire. A Brief Introduction to Boosting. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann Publisher, San Fransisco, 1999.

[8] J. Schürmann. *Polynomklassifikatoren für die Zeichenerkennung*. Oldenburg Verlag, München Wien, 1977.

[9] J. Schürmann. *Pattern Classification*. John Wiley&Sons, New York Chichester Brisbane Toronto Singapore, 1996.

[10] M. Skurichina. *Stabilizing Weak Classifiers*. Phd theses, Delft University of Technology, Delft Netherlands, 2001.

[11] C.Y. Suen and L.J. Lam. Multiple Classifier Combination Methodologies for Different Out Levels. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems Lecture Notes in Computer Science, Vol. 1857 52-66*. Springer-Verlag, Berlin Heidelberg New York, 2000.