The design of efficient pattern recognition systems

by

Lev Goldfarb

School of Computer Science

University of New Brunswick

Fredericton, NB E3B 5A3

Canada

ABSTRACT

A new general methodology for design of pattern recognition systems is proposed. This methodology is based on the new approach to pattern recognition proposed by the author. The principles of the design considered here should also be of use in the design of some complex database systems and error-correcting systems.

KEYWORDS

Pattern recognition, distance function, abstract nearest neighbour, search algorithm in metric space, pseudoeuclidean space

1. INTRODUCTION

During the last ten years, when the work on robotics and pattern recognition has progressed from small experiments to real problems, the importance of efficient design has become quite clear. In this short paper a very general underlying principle for design of pattern recognition systems is proposed. This principle has crystallized only after the development by the author of a general approach to pattern recognition [Goldfarb (1984a)], the appoach which was motivated by the unification of the two basic approaches to pattern recognition--syntactic and geometric (vector space) [Goldfarb (1984a), Goldfarb (1984b)].

The developed approach provides a general basis for making the decision which should depend only on a finite set of representative patterns independent of the chosen form of the pattern representation. As a result, it has become clear that the same framework allows one to design efficient systems for a variety of pattern recognition problems.

In this paper we restrict ourselves to the systems based on the following general principle: given an input object o (e.g. an image) the system has to choose a stored prototype object o_i (or several of them) which is a best match for the input object. In other words, we consider systems based on the abstract nearest-neighbour "philosophy". It goes without saying that such systems could be part of larger knowledge base systems.

2. DISTANCE FUNCTION AS A UNIFYING BASIS FOR GENERALISED BEST MATCH PROBLEMS

To rephrase the objective of the systmes we are concerned with one can say that we want to know how to design an efficient software for a general search problem when the representaion of the input element may contain some error, ie, how to design a system for a general best match problem.

Since I have argued elsewhere [Goldfarb (1984a), ch. 1,2] that the concept of a pairwise dissimilarity (distance) measure is a very universal and flexible tool for pattern recognition problems, it suffices to discuss here the present state of the affairs in regard to the acceptance of the distance measure as a basis for formulating pattern recognitin problems. The situation is quite ambivalent. On the one hand this fact (stated in italics) has been to some degree recognized--thus for example, 7th International Conference on Pattern Recognition had a separate session (Session 1.1.4) devoted to Similarity and Distance. On the other hand, this recognition has not resulted in a widespread acceptence of a general distance measure (ie, defined on non-vector representation) as a basis for solving PR problems--thus, for example, many problems considered in papers presented at the same conference although naturally fit into the "distance model" were not even formulated as such.

What are the reason which conditioned this ambivalent situation? There are several of them. The first and the most important one is a result of isolation of many applied areas from the mathematical developments of the last century. To be more specific, while the concept of a metric space was introduced in 1906 and has played a fundamental and unifying role in all the major branches of mathematics, these developments, unfortunately, have not yet affected many applied curriculums. As a consequence, the concept of a distance function remains for most, at best, just an abstract definition without any connections with the classical mathematics. The second reason comes form the fact that no efficient classification or nearest neighbour search methods in metric spaces have been found [see Feustel and Shapiro (1982)]. The approach proposed in Goldfarb (1984a) was developed, to an extent, to remove the latter as a relevent objection.

The above approach is based on the fact that there is an algorithm which for any finite metric space (P,Π), where Π is a distance function, constructs a <u>vector representation of</u> (P,Π). More precisely, by a vector representation of (P,Π) we shall mean a distance preserving (isometric) mapping of (P,Π) into a vector space. It is well known that in order to be able to measure distances in a vector space, the vector space should be supplied with an inner (scalar) product. It turns out that the class of

Euclidean inner product spaces is not large enough to "accommodate" any finite pseudometric (or even metric) space, necessitating consideration of a larger class of inner product vector spaces--the pseudoeuclidean spaces [Goldfarb (1984a) ch. 3].

The most known example of such a space is a 4-dimensional Minkowski space of special relativity theory. In general, a pseudoeuclidean space $R^{(n^+, n^-)}$ of dimension $n = n^+ + n^-$ is "contructed" from two non-commensurable Euclidean spaces R^{n^+} and R^{n^-}, exactly as Minkowski space is formed by the 3-dimensional space of "space-like" vectors and 1-dimensional space of "time-like" vectors [Greub (1974), p. 281]. The geomtry of pseudoeuclidean spaces is relatively well understood [Goldfarb (1984a) ch. 3; Greub (1974) ch. 9 sections 4 and 5], but the exposition of it will take us far afield.

For our purpose it suffices to note that

1. there exists a very efficient and reliable algorithm for constructing the vector representation [Goldfarb (1984a) ch. 6],
2. most of the decision-theoretic algorithms available in a Euclidean space can be directly generalized to pseudoeuclidean space [Goldfarb (1984a)],
3. the whole approach is remarkably adaptable to parallel architectures.

Thus, in view of the fact that metric configuration can easily be reconstructed in a vector space, the crucial advantage of the abstract metric representation over a Euclidean lies in the ability of the distance function to "move" the classes involved away from each other. The latter is a result of the fundamentally different modelling processes. While the classical vector representaion consists in lining up some numerical characteristics of a single object, the metric approach is based on a chosen dissimilarity measure defined on pairs of objects. As a consequence, the metric approach can benefit not only from the quantitative information in the chosen representation, but also from the structural information, local as well as global. In addition, the metric function itself can accentuate these class differences by adapting an appropriate weighting scheme [eg, weighted Levenshtein distance, Fu (1982) p. 248].

The advantages of the new approach over the syntactic approach (ie, the approach based on syntactic parsing) is considered elsewhere [Goldfarb, Bhavsar and Chan (1984)].

3. OUTLINE OF THE DESIGN

Let us assume now that given a pattern recognition problem we somehow managed to choose the right representation and the right distance functon. In addition, we have a training sample which sufficiently well represents the classes

involved. The main question we are concerned with is the design of an efficient scheme for classifying an input pattern p to one of the several classes C_j represented by a set of patterns P_j.

Although in the classical vector approach the solution has not completely crystallized, the appropriate design is emerging as consisting of two principal stages: 1) editing and condensing, 2) an efficient search algorithm together with the special data structure designed to facilitate the problem. To get some idea about the two stages the reader is refered to Devijver and Kittler (1982), section 3.11-3.14, Devijver (1981), section 5.3 (for the first stage), and to Friedman, Bentley and Finkel (1977), Fukunaga and Narendra (1975), Yunk (1976), Dewdney (1977) (for the second stage). Editing and condensing is supposed to reduce the training sample by removing the points which do not effect essentially the decision surface, while the search algorithm is based on construction of an efficient search tree by appropriately partitioning the vector space, which further reduces the number of the distance computations.

It is crucial to understand that the advances made in each of the above two stages were possible only because of the presence of the vector space structure, ie, without the assumption that patterns are represented in a Euclidean space practically no progress would have been possible at any stage. Thus, as in the classical statistical approach,

the presence of the vector space structure allows an efficient solution of the best match problem (in fact, considerably more efficient than that offer by the syntactic approach).

Now, since the new approach allows one to transfer the original more abstract problem (formulated in a metric space) from the metric space to the vector space, we have again at our disposal the efficient algorithms mentioned above. Thus, starting with any pattern representation we can still use all of the efficiency improvements afforded by the developments in the above two stages.

There are several major routes one can take in designing the sofeware for the two stages of a general pattern recognition system. A few preliminary notations are in order. Let $P = \{p_i\}_{i=0}^{k}$ be the training sample consisting of the subsets P_j representing pattern classes C_j, and let Π be the chosen distance function (metric) defined on P, so that (P, Π) is a finite metric space. Let

$$\alpha : (P, \Pi) \rightarrow R^{(n^+, n^-)} \qquad n = n^+ + n^-$$

be a vector representation of (P,Π) [see Goldfarb (1984a), Def. 4.1], ie, vectors $a_i = \alpha(p_i)$ have the same interdistance matrix (computed in the pseudoeuclidean space $R^{(n^+, n^-)}$) as that of the metric sapce (P,Π). We shall assume now that the dimension n of $R^{(n^+, n^-)}$ is a reduced dimension. Although the general questions related to the dimensionality reduction were considered in Goldfarb

(1984a), ch. 6 and section 7.1, this topic will be treated more comprehensively in a separate paper.

The firt approach to the software design is based on the method of abstract orthogonal projection [Goldfarb (1984a), section 7.2], which finds a "projection" of an input object p on the previously contructed space $R^{(n^+, n^-)}$:

$$\pi(p) = A_{\emptyset} \cdot G^{-1} \cdot b,$$

$A_{\emptyset}$ is the matrix whose columns are the coordinates of a_i, $G = G(a_1, \ldots, a_n)$ is the Gram matrix for the basis $\{a_m\}_{m=1}^{n}$ and b is the vector with the coordinates

$$b^m = \frac{1}{2} \left[\Pi^2(p, p_0) + \Pi^2(p_{i_m}, p_0) - \Pi^2(p_{i_m}, p) \right],$$

$$\alpha(p_{i_m}) = a_m, \qquad \alpha(p_0) = 0.$$

This "projection" is then considered as a vector representation of p in $R^{(n^+, n^-)}$ and the above two stages can be applied in the vector space $R^{(n^+, n^-)}$ The only overhead in this method in comprison with its classical counterpart is the computaion of the projection, which involves the computation of n+1 metric distances (p, p_{i_m}), $0 \le m \le n$, from p to some fixed elements of P, the computaion of the vector b and of the matrix product. It is important to note that the computation of the metric distances can be performed in parallel (using n+1 identical processors). The computation of vector b and of the two matrix products can also be implemented on a parallel computer in O(log n) time, where we have a considerable control over the dimension n of

the representation space, and therefore does not contribute significantly to the projection cost. Thus as one can see the cost of the on-line vector representation of an input pattern on a parallel architechture is dominated by the cost of computing a single distance, which in tern can also be computed in parallel.

It is important to understand that the above method allows one to reduce the number of processors in the brute-force nearest neighbor algorithm from k+1 (with k+1 elements in the orginal sample, which could run into 10^5) to n+1, where n with the appropriate choice of the distance function is equal to some small constant times the number of classes involved. This alone puts the hardware cost within reasonable limits.

The above approach is very reliable if the training sample is sufficiently large or sufficiently well approximates the underlying class distributions. If this is not feasible, or if the number of classes is very large, which results in a very high dimension n of the vector representation space, another scenario for the system design is possible along the following lines: divide the set of all classes into groups, construct the vector representation of each group of classes separately, apply the above design to each group, and then merge the resulting decision trees together.

In general, at the very beginning of the design process it shold be very useful to constuct a complete vector representation of the entire training sample in some low dimensional vector space, and to check if the pattern classes are well separated, since their separation can be improved by tuning the original distance function. In some cases each class could be satisfactory represented by one element [see Goldfarb and Chan (1984a) and (1984b)], and in other cases each class could be adequately represented by vertices of the piecewise linear class boundary, or, in some cases by vertices of the convex hull. To get an idea about the reduction in number of distance computaions achieved in the latter case the reader is refered to the several theorems of Renyi and Sulanke quoted in Shamos (1976), p. 275.

4. CONCLUSION

A new approach to the design of pattern recognition systemns, as well as others abstract search problems, is proposed. The approach does not impose any restrictions on the form of data representaion, and could be applied, in particular, to syntactic patterns. The design scheme relies on a new approach to pattern recognition [Goldfarb 1984a)]. The model is based on the concept of abstract distance measure defined on the samples, and allows one efficiently utilize the information contained in the interdistance

matrix by constructing an optimal vector representaion of the original sample in a pseudoeuclidean vector space which is completely determined by the input data. This in turn allows application of the efficient algorithms and data structures developed for search probelems in the Euclidean space.

REFERENCES

Goldfarb, L. (1984a). A new approach to pattern recognition. In: L.N. Kanal and A. Rosenfeld, eds., Progress in Pattern Recognition vol. 2 (in press). North-Holland Publishing Company, Amsterdam.

Goldfarb, L. (1984b). A unified approach to pattern recognition. Pattern Recognition (in press).

Feustel, C.D. and L.G. Shapiro (1982). The nearest neighbor problem in an abstract metric space. Pattern Recognition Letters 1, 125-128.

Greub, W. (1974). Linear Algebra. Springer.

Fu, K.S. (1982). Syntactic pattern recognition and applications. Prentice-Hall, Englewood Cliffs.

Goldfarb L., V.C. Bhavsar and T.Y.T. Chan (1984) On a more efficient alternative to the syntactic approach. Pattern Recognition Letters (submitted for publication).

Devijver, P.A., and J. Kittler (1982). Pattern recognition: a statistical approach. Prentice-Hall.

Devijver, P.A. (1981). Advances in non-parametric techniques of statistical pattern classification. In J. Kittler, K.S. Fu, and L.F. Pau, eds., Pattern Recognition Theory and Applications. Reidel, Dordrecht, Holland.

Friedman, J.H., J.L. Bentley, and R.A. Finkel (1977). An algorithm for finding best matches in logarithmic expected time. ACM Trans. Mathematical Sofeware 3, no. 3 (Sept. 1977), 209-226.

Fukunaga, K. and P.M. Narendra (1975). A branch and bound algorithm for computing k-nearest neighbors. IEEE Trans. Computer C-24, 750-753.

Yunk, T.P. (1976). A technique to identify nearest neighbors. IEEE Trans. Systems, Man and Cybernatics 6, 678-683.

Dewdney, A.K. (1977). Complexity of nearest neighbor searching in three and higher dimensions. Tech. Rept. No. 28, Dept. of Computer Science. University of Western Ontario, Canada.

Goldfarb, L. and T.Y.T. Chan (1984a). On a new unified approach to pattern recognition. Proc. of 7th International Conference on Pattern Recognition, 705-708.

Goldfarb, L. and T.Y.T. Chan (1984b). Improving syntactic approach. Pattern Recognition Letters (submitted for publication).

Shamos, M.I. (1976). Geometry and statistics: problems at the interface. In J.F. Traub, ed, Recent results and new directions in algorithms and complexity. Academic Press, New York.