International Journal of Soft Computing 3 (4): 288-292, 2008
ISSN: 1816-9503

Neuro-Fuzzy Approach for Pattern Recognition

T. Jebarajan, S. Arumugam, M.C. Loraine Charlet Annie
and D.C. Joy Winnie Wise
Deptartment of Computer Science and Enggneering,
Noorul Islam College of Enggneering, Kumaracoil, Thuckalay, Kanyakumari Dt.,
Tamilnadu, India

Abstract: An efficient and robust Pattern recognition algorithm is proposed in this study. The study makes an analysis on the popular classification and recognition algorithms, understanding their drawbacks adds features to the Fuzzy C-means classification algorithm, employs a feature extraction algorithm using wavelet and a hierarchical matching strategy in the recognition phase so as to accelerate matching process.

Key words: Clustering, matching, objects, pattern recognition, fuzzy, c-means, k-means

INTRODUCTION

The conceptualization of things, or objects, as belonging to classes is at the core of all knowledge. Once the classes are in the mind of the beholder, there are often many possible attributes and ways to classify a set of objects. Pattern recognition involves conceptualizing unknown things, or objects, as belonging to classes, which are stored at knowledge base. The knowledge base may often contain many possible attributes and ways to classify the set of objects. Pattern recognition involves two processes: classification, where a sample from a population of objects is partitioned into groups called classes and recognition, where a given unknown object from the same population is recognized as belonging to one of the established classes. A classification process examines a sample of objects that represents a population of such objects and partitions it into subsets (the classes) according to similarity of the objects within classes and dissimilarity of the objects between classes. This type of process is called self-organization, unsupervised learning, or clustering of the sample into clusters (classes or subclasses). On the other hand, once a process has clustered the sample into classes and each object is assigned a class label-an index value (or codeword) that designates the particular class-a recognizer can be trained to assign a class label to any unknown object from the same population (pattern recognition). The training process is called supervised learning or training of the recognizer. A trained recognizer can perform pattern recognition online.

Great strides have been made in computer vision to improve both the accuracy and speed of object recognition systems. Though, there exist systems that attain fast or even real-time performance with high detection rates (Kaufman *et al*., 1990; Leibe *et al*., 2003; Lowe, 2004; Papageorgiou *et al*., 2000; Triesch, 2001; Vidal *et al*., 2003; Wiskott *et al*., 1997), they are typically restricted to at most a few objects. In this study, we concentrate on a more robust and efficient classification and recognition algorithms to give the best Pattern Recognition. The research also is making an analysis of the available algorithms for classification and recognition, understanding their drawbacks is giving better efficiency in the proposed algorithm.

CLASSIFICATION ALGORITHMS

A classification of a population of objects involves partitioning it into subsets (classes) according to similarity of the objects within classes and dissimilarity of the objects between classes. There have been many classification algorithms in the history of digital image processing; the features of important classification algorithms are analyzed in my work and the best of which are applied in the classification algorithm of our work. The following are the classification algorithms analyzed in the work.

Classification by k-means algorithm: The k-means clustering algorithm for classification is the simplest, involves the classification of a set of Q-tupled feature

Corresponding Author: T. Jebarajan, Deptartment of Computer Science and Enggneering, Noorul Islam College of Enggneering, Kumaracoil, Thuckalay, Kanyakumari Dt., Tamilnadu, India

vectors into some k clusters by finding a cluster center such that the dissimilarity measure (Euclidean distance between the input feature vector and the cluster center) is minimized. The K-means algorithm is given:

Step 1: Randomly order the Q-feature vectors and input K (the number of classes).

Step 2: Select the first K of the Q-feature vectors as seeds (initial prototypes of classes).

Step 3: Assign each of the Q-feature vectors to the nearest prototype to form K classes (use the index c(q)=k to designate that $x^{(q)}$ belongs to Class k and count cluster sizes with s(k), which is incremented every time a vector is assigned to Class k).

Step 4: Average the feature vectors in each class to find K new centers. To average all vectors in Class k, use the following technique, starting with a(n)(k) = 0.0, where a(n)(k) is the value of component n of the Class k average vector and x(n)(q) is the nth component of $x^{(q)}$:

```
for k = 1 to K do //For each Class k:
for n = 1 to N do a(n)(k) = 0.0;
//initialize averages
for q = 1 to Q do
//For given Class k,
find
if (c(q) == k) then
//all vectors in Cluster k
for n == 1 to N do
//and for each component n,
a(n)(k) = a(n)(k) + x(n)(q);
//sum that component
if (s(k) > 1) then
a(n)(k) = a(n)(k)/s(k); //Average Cluster k
```

Step 5: If ((not first pass) and (no class has changed)) then exit, else go to Step 3 above.

Many algorithms use the k-means to start but apply some adjustment to aid with the seeding. They showed that random drawing of many more seeds worked better than the use of the first K-feature vectors. Selim *et al.* (1984) showed that the algorithm converges to a local minimum in the sum-squared error.

However, Looney (1997) showed that there is no guarantee of optimality of the clustering, which depends on K and the K initial seeds. There are also clustering algorithms that calculate the centers each time a vector is added to the cluster. Kaufman *et al.* (1990) optimized the least mean square error by using median prototypes. A clustering is good if the clusters are relatively compact (packed closely about the center) and relatively well separated. The number of clusters K must therefore reorder the Q-feature vectors and repeat the clustering several times to find the best clustering. The optimized cluster selection based on a cluster validity measure has also been worked through the algorithm of Xie *et al.* (1991). But the problem while using k-means clustering algorithm is that it provides only a partial optimum solution and computational burden due to its late convergence. The importance of all these algorithms is analyzed and to be applied in my work.

Fuzzy C-means clustering: Bezdek (1973) proposed the fuzzy C-means as an improvement over the k-means clustering. FCM partitions a population of feature vectors into fuzzy groups, by specifying membership grades (0-1) to show the belongingness of each data point on all groups. The fuzzy C-means algorithm is:

Step1: Initialize the membership matrix U with random values between 0 and 1 such that the constraints in Eq. (1) are satisfied.

$$\sum_{i=1}^{c} u_{ij} = 1, \forall j = 1...n \tag{1}$$

Step 2: Calculate c fuzzy centers c_i, i=1...c using Eq. (2).

$$c_i = \frac{\sum_{j=1}^{n} u_{ij}^{m} x_j}{\sum_{j=1}^{n} u_{ij}^{m}} \tag{2}$$

Step 3: Compute the cost function according to Eq. (3).

$$J(U, c_1, ..., cc) = \sum_{i=1}^{c} Ji = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} d_{ij}^{2} \tag{3}$$

Stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold.

Step 4: Compute new membership matrix. Using Eq. 4.

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}} \right)^{2/m-1}} \tag{4}$$

Go to Step 2.

The fuzzy C-means algorithm is better than the k-means algorithm, so my classification algorithm is looking towards optimizing FCM using features of certain soft computing techniques.

RECOGNITION

Given a set of exemplar features, recognition process assigns a class label to any incoming input feature vector of the same population by a supervised training algorithm. There are many methods that train a recognition process via supervised learning for online recognition of an incoming stream of feature vectors from the population of interest. The following are the recognition algorithms analyzed in the work.

Probabilistic neural network algorithm: A Probabilistic Neural Network (PNN) has three layers of nodes that recognize K classes. The input layer contains N nodes: one for each of the N input features of a feature vector. These are fan-out nodes that branch at each feature input node to all nodes in the hidden layer so that each hidden node receives the complete input feature vector x. The hidden nodes are collected into groups, one group for each of the K classes, each hidden node in the group for Class k corresponds to a Gaussian function entered on its associated feature vector in the kth class. All of the Gaussians in a class group feed their functional values to the same output layer node for that class, so there are K output nodes.

At the output node for Class k, all of the Gaussian values for Class k are summed and the total sum is scaled to force the integral of the sum to be unity so that the sum forms a probability density function. Here we temporarily use special notation for clarity. Let there be P exemplar feature vectors $\{x^{(p)}: p = 1\ldots P\}$ labeled as Class 1 and let there be Q exemplar feature vectors $(y^{(q)}: q = 1\ldots Q)$ labeled as Class 2 (Fig. 1). In the hidden layer there are P nodes in the group for Class 1 and Q nodes in the group for Class 2. The equations for each Gaussian centered on the respective Class 1 and Class 2 points $x^{(p)}$ and $y^{(q)}$ (feature vectors) are:

$$g_1(x)=[1/\sqrt{2}\prod\sigma^2]\exp\{-\|x-x^{(p)}\|^2/(2\sigma^2)\} \quad (5)$$

$$g_2(x)=[1/\sqrt{2}\prod\sigma^2]\exp\{-\|y-y^{(q)}\|^2/(2\sigma^2) \quad (6)$$

The s values can be taken to be one half the average distance between the feature vectors in the same group, or at each exemplar it can be one half the distance from the exemplar to its nearest other exemplar vector. The kth output node sums the values received from the hidden nodes in the kth group, called mixed Gaussians or Parzen windows. The sums are defined by:

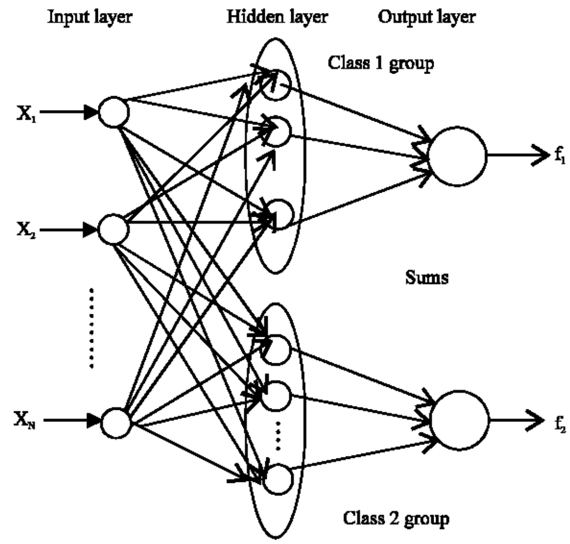

Fig. 1: A probabilistic neural network

$$f_1(x)=[1/\sqrt{(2\prod\sigma^2)^P}]\,(1/P)\sum_{(p=1,P)}\exp\{-\|x-x^{(p)}\|^2/(2\sigma^2)\} \quad (7)$$

$$f_2(y)=[1/\sqrt{(2\prod\sigma^2)^P}]\,(1/Q)\sum_{(q=1,Q)}\exp\{-\|y-y^{(q)}\|^2/(2\sigma^2) \quad (8)$$

Any input vector is put through both functions and the maximum value (maximum a posteriori, or MAP value) of f_1 and f_2 decides the class. For $K > 2$ classes the process is analogous. There is no iteration or computation of weights. For a large number of Gaussians in a sum, the error can be significant. Thus, thinning those may reduce the feature vectors in each class that are too close to another one.

Fuzzy neural networks: The Fuzzy Neural Networks (FNNs) are similar to the PNNs. Let there be K classes and let x be any feature vector from the population of interest to be recognized. The Class k exemplar feature vectors are denoted by $x^{(qk)}$ for qk = 1… Q_k. We replace the functions of Eq. (9) and (10) with the following more simply scaled sums.

$$f_1(x) = (1/Q_1)\sum_{(p=1,p)}\exp\{-\|x-x^{(q1)}\|^2/(2\sigma^2)\} \quad (9)$$

$$f_k(x) = (1/Q_k)\sum_{(q=1,Q)}\exp\{-\|x-x^{(qk)}\|^2/(2\sigma^2)\} \quad (10)$$

These functions are the K fuzzy-set membership functions whose functional values are the relative fuzzy truths of memberships in the K respective classes. Thus, x belongs to the class with the highest fuzzy value. When there is a clear winner, then x belongs to a single class, but otherwise it may belong to more than one class with the given relative fuzzy truths. No training of weights is required. The exemplar vectors may be thinned as previously done.

The recognition of individual objects with complete shapes has been studied for a long time, can be handled without much difficulty with many existing techniques, such as shape signature, moments and Fourier descriptors. However, problems arise when the object is partially occluded. Those aforementioned global techniques are incapable to solve occlusion problem. This problem has significant importance in an industrial environment and robot recognition. Hence, it has attracted plenty of interests since 1980s. Many structural methods have been reported. Some researchers using dominant-point based recognition methods. However, dominant points alone are insufficient to form a complete integrated representation of an object. Polygon representation is another branch. However, this method has the drawback that it is unstable in finding break points for non-polygonal objects. Some researchers used one or the combination of some basic geometric features, such as line, arc, corner and end to describe a contour but some complex object may not be fully represented only by the basic geometric features accurately. Spectral descriptors overcome the problem of noise sensitivity and boundary variation by analyzing shape in spectral domain. Gorman proposed a partial shape recognition technique utilizing local feature described by Fourier descriptors. However, this approach needs approximated scale information to set the threshold for curve partitioning. Other than Fourier descriptor, Wavelet descriptor is another spectral descriptor. It has the advantage of multi-resolution approximation in both spatial and spectral domain over Fourier descriptor. Some pattern recognition approaches using wavelets have been reported. However, approaches in focus on no-occluded case which rely the global information, hence they are inapplicable to solve occlusion problems. Yoon *et al.* (1998) proposed a notion of object representations using curvature zero-crossings of the approximation of object contour. The study has shown the validity, efficiency and reliability of the representations and also shown the representation's capability of solving scaling and occlusion problem. However, match of the zero-crossings of the object contours between target and model is insufficient to confirm the object in scene is matched with the particular model. From the above literature review, we have shown that the approaches using traditional methods have their own drawbacks and wavelet is a promising tool for recognition. However, recognition of partial occluded object using wavelet techniques is still a challenging task.

In this study, we will represent a pattern recognition approach including a feature extraction algorithm using wavelet and a hierarchical matching strategy which can accelerate matching process.

SYSTEM DESCRIPTION

Pattern recognition approach proposed in this study comprises three stages: clustering, object representation and recognition. In the clustering stage, a population of objects is classified into classes based on the improved Fuzzy C means algorithm. In the object representation stage, features are extracted from counter of an object through four steps:

- Preprocessing step which involves image enhancement, noise removal, edge detection and boundary tracking
- Segmentation of object boundaries into independent segments.
- Normalization step wherein segment is made translation invariant, rotational invariant, scaling invariant.
- Wavelet decomposition step wherein the normalized coordinates of the segments are decomposed into three levels by B-spline wavelet. In the recognition stage, an object in the scene is recognized as one of the models stored in the database.

MATCHING ALGORITHM

Our pattern recognition system is model based, it is to recognize an object in scene as one of a number of models whose representation are known and stored in database. The matching process consists of two steps: segments matching and hypothesis verification.

Segments matching: Firstly, we try to match the feature matrix of segments between object in scene and models iteratively to find matched segment-pair if any. Matching in such iterative manner can be very computation expensive, especially when the number of models in database is huge. From wavelet reconstruction theory, we can use to scaling coefficient to reconstruct an approximation of original segment and better and better approximation can be obtained by adding wavelet coefficients at all levels and on. Hence, our feature matrix forms a multiresolution representation and we match approximation image at the final state first, if the dissimilarity function is less than a proper chosen threshold value, the matching process will proceed to higher resolution, otherwise just simply reject this candidate. Similarly, we match the entire detailed coefficient in the same manner.

By the hierarchical matching strategy, we can eliminate most of ineligible candidates in earlier stages, so that the matching speed can be accelerated dramatically.

Hypothesis verification: In the segment matching process, if a set of reasonable matching pairs occurs then a possible hypothesis that the model object appears in the scene image, the next step is to verify the hypothesis. Because a set of reasonable matching segments does not guarantee the match of objects for some special cases, such as the matching between rectangle and square.

If more than one matched segment-pair candidates are found in segment matching step, the validity of the aforementioned fact of matched segment-pair candidates will be checked to eliminate false matching. The relative scale and orientation of the segment-pairs can be calculated using the similarity transformation information stored earlier during normalization with ease. There are the following possibilities of matching result:

- All segments of object in scene are matched with all segments of model, respectively and their relative scale and orientation are same. For this case model is presented in scene and without occlusion.
- More than one (but not all) segments of the object in scene are matched with some segments of model, respectively and their relative scale and orientation are same. For this case, the model is presented in scene but with occlusion. Those unmatched segments of model are occluded.
- Only one matched segment-pair is found. For this case, the validity of the fact can not be checked. Hence, the model may present in scene.
- More than one segments of the object in scene are matched with some segments of model, respectively, but some of their relative scale and orientation differ. For this case, false matching should be eliminated. The largest cluster of segment-pair candidates which have same relative position, scale and orientation implies true.

CONCLUSION

The proposed pattern recognition is aimed at avoiding the problems associated with traditional methods by using fuzzy c-means clustering algorithm instead of k-means clustering algorithm for object recognition since c-means is faster and efficient than k-means clustering algorithm. As a second step the algorithm uses a neural network based approach to the classifier design due to its compactness, efficiency and automatic operations there by avoiding problems associated with similarity measure calculations within a predefined block.

REFERENCES

Bezdek, J.C., 1973. Fuzzy Mathematics in Pattern Classification. Ph.D. thesis, Center for Applied Mathematics, Cornell University, Ithaca, NY.

Kaufman, L. and P. Rousseeuw, 1990, Finding Groups in Data: An Introduction to Cluster Analysis. Text Book, John Wiley and Sons, NY.

Leibe, B. and B. Schiele, 2003. Analyzing appearance and contour based methods for object categorization. In: Proc. IEEE Conf. Comput Vision Pattern Recogn. (CVPR), 2: 409-415.

Looney, C.G., 1997. Pattern Recognition using neural networks: Theory and algorithms for engineers and scientists. Text Book, Oxford University Press, NY.

Lowe, D., 2004. Distinctive image features from scale-invariant key points. Int. J. Comput. Vision, 60 (2): 91-110.

Papageorgiou, C. and T. Poggio, 2000. A trainable system for object detection. Int. J. Comput. Vision, 38 (1): 15-33.

Selim, S.Z. and M.A. Ismail, 1984. k-means type algorithms: A generalized convergence theorem and characterization of local optimality. IEEE. Trans. Pattern Anal. Machine Intell., 6: 81-87.

Triesch, J., 2001. A system for person-independent hand posture recognition against complex backgrounds. IEEE. Trans. Pattern Anal. Machine Intell., 23 (12): 1449-1453.

Vidal-Naquet, M. and S. Ullman, 2003. Object recognition with informative features and linear classification. In Proc. Int. Conf. Comput. Vision (ICCV) Nice France, pp: 281-288.

Wiskott, L., J.M. Fellous, Kr¨uger and C.V. Malsburg, 1997. Face recognition by elastic bunch graph matching. IEEE. Trans. Pattern Analysis and Machine Intelligence, 19 (7): 775-779.

Xie, X.L. and G. Beni, 1991. A validity measure for fuzzy clustering. IEEE. Trans. Pattern Anal. Machine Intell., 13 (8): 841-847.

Yoon, S.H., J.H. Kim, W.E. Alexander, S.M. Park and K.H. Sohn, 1998. An optimum solution for scale-invariant object recognition based on the multiresolution approximation. Pattern Recog., 7(30): 889-908.