Application of Fuzzy Network to Air -target Recognition

Du Gan Zhang Shouhong
Key Laboratory for Radar Signal Processing, Xidian University,
Xi'an Shaanxi 710071,P.R.China
Email address: dugan@rsp.xidian.edu.cn

Abstract: In this paper, the application of fuzzy self-organization network to air-target recognition is studied. On this basis, a new method to recognize the infrared images of airplane and its decoy is proposed. The fractal dimension (FD) and gray entropy are extracted from infrared images, then the Fuzzy *c*-Means Neural Network (FCMNN) is used to fuse the information in decision level. The experiments indicate that the new method is reliable and more effective.

Key words: neural network, target recognition

1 Introduction

The recognition of air-target and its decoy is one of the key subjects in infrared imaging guidance system. In the computer vision field, due to some factors such as imaging distance, direction and position, acquired images may be subjected to distortion (rotation, translation and etc.). So the features based on geometric invariation should be extracted. On the other hand, in decision level the features will fluctuate because of the imaging noise, so the uncertainty rules should be applied. Many researchers have used fuzzy mathematics to study target recognition in recent years. Liu[1] used fuzzy pyramid algorithm to solve the problem of invariant object recognition; Based[2] studied pattern recognition algorithm with fuzzy objective function. In this paper, the artificial network is used to study this kind of problem.

2 Theory of Fuzzy C-Means Neural Network

Artificial networks are nonlinear dynamical systems intrinsically. The key features of neural networks are asynchronous parallel processing, continuous-time dynamics, and global interconnection of network elements. The networks have great potentiality of studying the multi-dimension features space, which constructed by multi-feature.

Kohonen clustering networks (KCNs)[3] are well known for cluster analysis. Kohonen's work has become particularly timely in recent years because of the widespread resurgence of interest in the theory and applications of neural network structures. However, KCNs suffer from several major problems: KCNs are heuristic procedures, so termination is not based on optimizing any model of the process or its data. The final weight vectors usually depend on the input sequence. Different initial conditions usually yield different results. Several parameters of the KCNs algorithms such as the learning rate, the size of update neighborhood, and the strategy to alter these two parameters during learning, must be varied from one data set to another to achieve "useful" results. On the other hand, it is well known that KCN clustering is closely related to the *c*-Means (CM) algorithms. Since CM algorithms are optimization procedures, whereas KCN is not, integration of CM and KCN is one way to address several problems of KCNs. Fuzzy *c*-Means Neural Network (FCMNN)[4] which combines the Fuzzy *c*-Means algorithm with Kohonen clustering network, is a new network model. It is noted for fault tolerance, high speed and efficiency.

(1) Fuzzy *c*-Means Algorithms:

Let c be an integer, $1<c<n$, and let $\mathbf{X}=\{x_1,x_2,...x_n\}$ denote a set of n feature vectors in $\mathbf{R}^{\mathbf{P}}$, $U=[u_{ik}]_{c\times n}$ is a fuzzy c-partition of $\mathbf{X}$, $V=(v_1, v_2,...v_c)$ are cluster centers in $\mathbf{R}^{\mathbf{P}}$. The objective function for Fuzzy *c*-Means algorithms can be defined as:

$$J_m(U,V)=\sum_{i=1}^{c}\sum_{k=1}^{n}u_{ik}^{m}(\| x_k - v_i \|_A)^2 \qquad (1)$$

ST

$$\sum_{i=1}^{c}u_{ik}=1 \qquad \forall k=1,2,...n$$

$$0<\sum_{k=1}^{n}u_{ik}<n \qquad \forall i=1,2..c$$

$$u_{ik}\in[0,1] \qquad m\in[1,\infty)$$

where A is any position definite (P×P) matrix, and $\|x_k-v_i\|_A=(x_k-v_i)^T A(x_k-v_i)$ is the distance (in the A norm) from x_k to v_i. Optimal partition $\mathbf{U}^*$ of $\mathbf{X}$ are taken from pairs($\mathbf{U}^*$, $\mathbf{V}^*$) that are "minimizers" of J_m. Fuzzy c-Means algorithms can be describes as:

Step1. Fix: $1<c<n$, $m>1$, $\| \ \|_A$, and $\varepsilon>0$ some small positive constant.

Step2. Initialize network weight vector $V_0=(v_{1,0}, v_{2,0},...v_{c,0})\in R^{cp}$.

Step3. For $t=1,2,..t_{max}$:

a. Update all memberships $[u_{ik,t}]$:

$$u_{ik}=\frac{1}{\sum_{j=1}^{c}\left(\frac{\| x_k - v_i \|_A}{\| x_k - v_j \|_A}\right)^{2/(m-1)}} \qquad (2)$$

$$\forall i=1,2,..c \qquad k=1,2,..n$$

b. Update all weight vectors $\{v_{i,t}\}$:

$$v_i=\frac{\sum_{k=1}^{n}u_{ik}^{m}\cdot x_k}{\sum_{k=1}^{n}u_{ik}^{m}} \qquad (3)$$

$$\forall i=1,2..c$$

c. Compute E_t :

$$E_t=\| v_t - v_{t-1} \|^2=\sum_i \| v_{i,t} - v_{i,t-1} \|^2 \qquad (4)$$

d If $E_t<\varepsilon$ stop; else next *t*.

(2) Structure of Fuzzy *c*-Means Neural Network

FCMNN is an unsupervised scheme that finds the “best” set of weights for fuzzy clusters in an iterative, non-sequential manner. It is independent of the sequence of feed of the input data. FCMNN is a two-layer feed-forward propagation network. As shown in Fig1, the architecture is simple and consists of an input layer and an output layer.

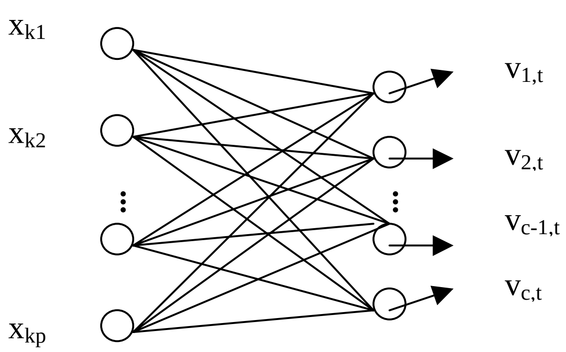

Fig1.The structure of a Kohonen clustering network

The input layer has p neurons (equivalent to the number of input data dimension). These neurons of input layer are linear neurons, and their transfer functions are liner function. Neurons of output layer are fuzzy competitive output neurons, and the number of neurons is *c* (equivalent to the desired number of classes). The output of second layer lies in the range [0,1] .The number of weight vectors which connects input layer and output layer is $p\times c$.

(3) Fuzzy *c*-Means Neural Network (FCMNN) Algorithm

FCMNN can be viewed as a Kohonen type of Fuzzy *c*-Means model, but it is "self-organizing" since the update of weigh vectors in the competitive layer are automatically adjusted during learning. It uses fuzzy competitive learning algorithm. Given an input vector, every weight vector updates its value, and the quantity of update of every weight vector is inversely proportional to its distance from the input vector.

The algorithm can be described as following:

Step1. Fix c, $\| \|_A$, and $\varepsilon>0$ some small positive constant. Fix m_1, m_0 and t_{max}= iterate limit.

Step2. Initialize $V_0=(v_{1,0}, v_{2,0},...v_{c,0})\in R^{cp}$, and $F_{i,0}=1$.

Step3. For t=1,2,..t_{max} ;For k=1,2,..n:

a. Compute $u_{ik.t}$:

$$if \quad I_k=\phi \quad then \quad u_{ik}=\frac{1}{\sum_{j=1}^{c}[\frac{d_{ik}^2}{d_{jk}^2}]^{1/m-1}} \qquad (5)$$

$$else \quad u_{ik}=0, \qquad \forall i\in \bar{I}_k\,, \quad \sum_{i\in I_k}u_{ik}=1 \qquad (6)$$

$$where \quad I_k=\{i\,|\,1\le i\le c, d_{ik}=0\}, \bar{I}_k=\{1,2..c\}-I_k$$
$$d_{ik}^2=\| x_k-v_i \|_A^2$$

b. Adjust parameter m_t:

$$m_t=m_0+(m_1-m_0)/t_{\max} \qquad (7)$$

c. Adjust learning rate $a_{i,t}$

$$F_{i,t}=F_{i,t-1}+u_{ik} \qquad (8)$$

$$a_{i,t}=1/F_{i,t} \qquad (9)$$

d. Update network weights:

$$v_{i,t}=v_{i,t-1}+a_{i,t}\cdot u_{ik}^{m_t}[x_k-v_{i,t-1}] \qquad (10)$$

Step4. Compute E_t :

$$E_t=\| v_t-v_{t-1} \|^2=\sum_i \| v_{i,t}-v_{i,t-1} \|^2 \qquad (11)$$

Step5. If $E_t<\varepsilon$ stop; else next *t*.

3 Features Extracting

In this paper, the fractal dimension (FD) and gray entropy are extracted from infrared images, then the FCMNN is used to fuse the information in decision level. The entropy is defined as:

$$h=-\sum_l p(l)\log_2 p(l) \qquad (12)$$

where p(l) is the probability that the lth gray value is occur.

We use the ε-blanket method of estimation of FD suggested by Mandelbrot[5]. In this extension, the image can be viewed as a hilly terrain surface whose height from the normal ground is proportional to the image gray value. Then all points at distance ε from the surface on both sides create a blanket of thickness 2ε. The estimated surface area is the volume of the blanket divided by 2ε. For different ε, blanket area can be iteratively estimated as follows. The covering blanket is defined by its upper surface u_ε and the lower surface b_ε. Initially, given the gray level function: g(i,j), $u_0(i,j)=b_0(i,j)=g(i,j)$, For ε=1,2,3.., the blanket surfaces are defined as follows:

$$u_\varepsilon(i,j)=\max\{u_{\varepsilon-1}(i,j)+1, \max_{d(i,j,m,n)\le 1} u_{\varepsilon-1}(m,n)\}$$
$$b_\varepsilon(i,j)=\min\{b_{\varepsilon-1}(i,j)-1, \min_{d(i,j,m,n)\le 1} b_{\varepsilon-1}(m,n)\} \qquad (13)$$

where d(i,j,m,n) is the distance between pixels (i,j) and (m,n).

Volume of the blanket is given by

$$v_\varepsilon = \sum_{i,j} (u_\varepsilon(i,j) - b_\varepsilon(i,j)) \qquad (14)$$

While the surface area is measured as

$$A(\varepsilon) = \frac{(v_\varepsilon - v_{\varepsilon-1})}{2} \qquad (15)$$

The area of fractal surface behaves according to the following expression:

$$A(\varepsilon) = F\varepsilon^{2-D} \qquad (16)$$

Fractal dimension can be derived from the least square linear fit of log-log plot A(ε) and ε , with the help of equation (16).

When the fractal feature of a target is extracted, to every pixel, compute the fractal dimension (FD) within a small neighborhood around it. After computing every pixel in the image, the average value of FD can be obtained.

4 Experiments and Conclusion

For our experiment we use 19 images of airplane and 14 images of its decoy. The images are quantised in 256 gray levels, each image is of size 128×128 pixels. The FD of each image is computed on overlapping windows of size 7×7. The entropy feature of each image is extracted too. In decision level, we use FCMNN to fuse features. Because of the difference of dimensions, two features are normalized. The results obtained from multi-feature fusion with those obtained from single feature are compared. In the first experiment, the original images are used, the correct recognition rate by fusion algorithm, FD feature and entropy feature is 94%, 88%, 82% respectively. In the second experiment, for the images on which the zero-mean Gaussian noise is added, the correct recognition rate by our approach for 33 images is 91%, 82%, 79%. Obviously, it is higher than the result obtained by any single feature. This experiment shows that the FCMNN method can efficiently improve the accuracy.

FCMNN, which integrates the Fuzzy c-Means model into the structure of the Kohonen network, is a promising clustering method. It can be used in many fields, such as image segmentation、target identification and so on.

References:

[1] Liu X,Tan S,Srinivasan V,OngS,Xie W,Fuzzy Pyramid-based Invariant Object Recognition, Pattern ecognition,1994,27:741-756.

[2] Based JC. Pattern Recognition with Fuzzy Objective Function Algorithms.New York: Plenum 1981

[3] T.Kohonen, Self-organization and Association Memory.Berlin:springer-rerlay 1984

[4] Li Wenhua, Fuzzy Clustering Method and Fuzzy Clustering Network, Journal of Xidian University,1996.1:23-27

[5] Mandelbrot B.B,The fractal geometry of nature, Freeman,New York 1982 .