Neural Processing Letters **18:** 81–95, 2003.

© 2003 *Kluwer Academic Publishers. Printed in the Netherlands.*

2-D Shape Recognition using Recursive Landmark Determination and Fuzzy ART Network Learning

APIWAT SAENGDEEJING, ZHIHUA QU, NOPPHAMAS CHAEROENLAP, and YUFANG JIN
School of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816, USA. e-mail: qu@pegasus.cc.ucf.edu

Abstract. In this Letter, 2-D shape recognition is done using a combination of recursive search of landmarks, landmark-based invariant features, and a fuzzy ART neural-network classifier. To make this novel combination work well, an upper limit is imposed on the number of total landmarks allowed, and this maximum size is then translated into fixed dimensions of invariant features and into the neural processing of the features. It is shown that the recursive landmark search approximates very well any smooth 2-D shape contour, that the shape features used are independent of perspective transformation, and that, when combined with a fuzzy ART classifier, unknown features can be efficiently learned on-line to identify multiple distinct objects. An illustrative example is used to demonstrate effectiveness of the proposed algorithm.

Key words. fuzzy ART network, recursive landmark determination, 2-D shape recognition

1. Introduction

Object recognition is one of the essential characteristics of an intelligent system, and its importance is clear in such areas as machine vision, inspection and automation, and target identification. For recognizing geometrically different objects, shape contour is the most obvious and important feature to be used. Although an object can be described well by its shape contour, the shape contour may change depending on camera perspective and/or motion of the object. Furthermore, in order to recognize several known objects and to learn unknown objects, various shape contours need to be compared and classified. To this end, an analytical shape representation has to be developed.

There have been two ways of modelling shape representations. In the spatial domain, methods such as, chain code [16], run length coding [7, 9], polygon approximation [6, 14], convex hull [10, 17] are proposed. In the transform domain, there are such techniques as Fourier Descriptor [8, 18], short-time Fourier Transform, Gabor, and wavelet transforms [1]. These methods have been studied by many researchers and have been used successfully in many applications. To accommodate changes in perspective and to guarantee successful recognition, invariant features can be developed to eliminate the impact of any affine transform [2, 3, 15]. It is known that, if the depth of an object is small compared with its distance to the camera, the object can be modelled as a 2-D object and its viewing and projection transformation becomes affine.

In this Letter, a new method of 2-D object identification is proposed for applications in robotics and automation [13]. The proposed method includes three components: a recursive search routine to find a limited set of best landmarks as shape features (that is, a constrained optimal polygon approximation), a new composite invariant feature that is calculated based on the landmarks found and is independent of reference points, and the application of fuzzy adaptive resonance theory [5]. This combination makes it possible to efficiently describe any shape contour (known or unknown), to extract transformation-independent features, and to learn and classify distinct objects in a set of still or time sequential images. Computational efficiency and real-time learning capability are the main advantages of the proposed method. An example of target identification is used to illustrate effectiveness of the proposed method.

2. Shape Acquisition and Processing

Upon completing edge detection, a specific perspective of a target to be identified is obtained, and it can usually be characterized as one closed contour or several contours. To illustrate the main idea, we assume that the perspective is described one closed contour.

If the camera is moving relative to the target, the perspectives will change over time and as long as the depth of the target can be neglected compared to the distance between the camera and the target, the target can be treated as 2-dimensional and its perspective changes can be described by an affine transformation [12].

To identify the specific target without knowing the transformation, one needs to determine a number of invariant features for the target. In the problem of identifying multiple targets, their corresponding invariant features need to be learned automatically. To this end, a multiple-object recognition technique is proposed in this Letter and it consists of the following modules:

- A group of landmarks will be chosen automatically through a recursive algorithm to characterize closed contours in a perspective of the targets.
- A sequence of transformation-invariant features will be extracted based on the landmarks associated with each closed contour.
- A fuzzy ART classifier is used to separate and learn the invariant features of various objects and hence to distinguish different objects.

3. Recursive Selection of Landmarks

In general, a curve (that is, a portion of a closed, shape contour) can be simply approximated by such pre-defined geometric primitives as line segments. This approach, traced back to [11, 14] and often called polygonal approximation in computer graphics, is used in the proposed recognition technique. Here, the polygonal approximation is applied to best capture the shape features in a perspective by searching for the maximum distances between pairs of points on the contour until a certain minimum resolution is reached and, to be computationally efficient, a maximum value is imposed on the number of vertices to be selected. Specifically, a recursive algorithm is proposed to locate vertices of a fitted polygon, called landmarks, by partitioning the contour according to the maximum distance found and by repeating the partition through a tree-like search.

To describe the proposed algorithm of locating landmarks, let

- X_p be the set of all points, p_i, on a digital image of the closed contour, i.e., $X_p = \{p_i, i = 1, \ldots, N_p\} = \{[x_i, y_i, 0]^T, i = 1, 2, \ldots, N_p\}$ such that point p_i is adjacent to both p_{i+1} and p_{i-1}.
- $\Omega \subset X_p$ be the set of landmark points to be found, that is, $\Omega = \{q_j, j = 1, \ldots, N_l\}$ where q_j is the jth landmark and N_l, to be found, is the dimension of set Ω. Typically, $N_l << N_p$.
- $D_{\max}$ is a positive integer representing the maximum search depth (or the maximum number of recursion steps, or maximum search level in the tree). Normally, $2^{D_{\max}+1} \geqslant N_l$.
- Ω_d be the set of landmark points found up to the dth iteration, where $d = 0, \ldots, D_{\max} + 1$. Thus, $\Omega_1 \subset \Omega_2 \subset \cdots = \Omega$. Assume without loss of generality that $\Omega_d = \{q_i^{(d)}, i = 1, \ldots, N^{(d)}\}$ be sorted such that landmark point $q_i^{(d)}$ is adjacent to both $q_{i+1}^{(d)}$ and $q_{i-1}^{(d)}$.
- $S_i^{(d)}$ be the set of points that are on the contour segment with its two end points at $q_i^{(d)}$ and $q_{i+1}^{(d)}$.

Then, the proposed recursive algorithm is as follows: given any positive constant ϵ and integer $D_{\max}$,

Initialization: Set $d = 0$, $N^{(0)} = 2$, and find $q_1^{(0)}$ and $q_2^{(0)}$ such that

$$\|q_1 - q_2\| = \max_{p_i, p_j \in X_p} \|p_i - p_j\|,$$

where $\|\cdot\|$ denotes the Euclidean norm.

Recursion at Level d: Set $\Omega_d = \Omega_{d-1}$. For each contour segment $S_i^{(d-1)}$, $i = 1, \ldots, N^{(d-1)}$, find q_i' such that

$$\|q_i^{(d-1)} - q_i'\| + \|q_i' - q_{i+1}^{(d-1)}\| = \max_{p \in S_i^{(d-1)}} [\|q_i^{(d-1)} - p\| + \|p - q_{i+1}^{(d-1)}\|].$$

If

$$\|q_i^{(d-1)} - q_i'\| + \|q_i' - q_{i+1}^{(d-1)}\| > \|q_i^{(d-1)} - q_{i+1}^{(d-1)}\| + \epsilon, \tag{3.1}$$

add q_i' into set Ω_d; otherwise, ignore q_i'.

Once segments $S_i^{(d-1)}$ are exhausted, set Ω_d is obtained after a simple sorting within itself. Let $d = d + 1$. If $d \leqslant D_{\max}$, continue the recursion; otherwise, exit the loop.

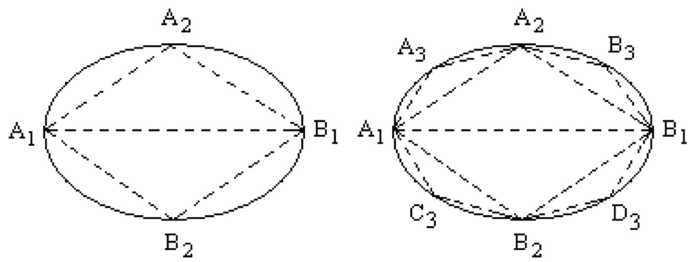

Figure 1. Recursive assignment of landmark locations.

As an example, consider the closed contour in Figure 1. During the initial step, two entry-level landmarks A_1 and B_1 are found by searching for the maximum distance between the points on the contour. These two landmarks partition the shape contour into two segments. Using the proposed algorithm, one can recursively locate landmark points at different levels (denoted by A_i, B_i, C_i, and so on for $i = 2, \ldots$) by finding the maximum accumulated distance (i.e., maximizing the sum of the two distances between a point in a segment to the end points of the segment). Note that the landmarks found in the previous steps are the ending points of various segments in the current recursion. In doing so, landmark points A_2 and B_2 are found in the level-1 search, and landmark points A_3, B_3, C_3 and D_3 are found in the level-2 search, and so on. This process continues until either the maximum search level is reached or the maximum accumulated distance becomes less than ϵ. It is obvious from this recursive mechanism that, given a relatively large $D_{\max}$ and a relatively small ϵ, the whole set of landmarks can approximate any smooth shape contour. In fact, the proposed recursion represents the process of a constrained optimal approximation.

To illustrate the proposed landmark search algorithm, consider the shaded region in Figure 2. A simple edge detection algorithm can be readily applied to find its shape contour. Upon completion, the proposed landmark search algorithm is carried out, the results and corresponding polygon segments are shown in Figure 2, and

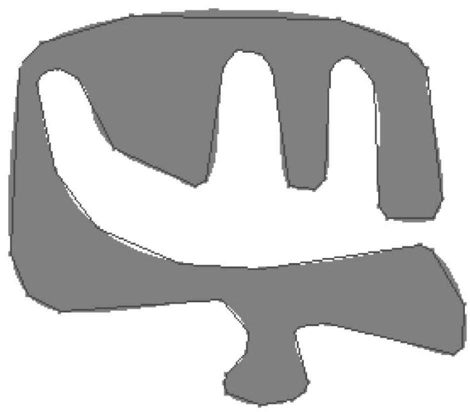

Figure 2. An arbitrary shape and the corresponding results of landmark search.

effectiveness of the algorithm is apparent. In this example, $D_{\max} = 8$ and $\epsilon = 1$ are set, and the proposed algorithm locates 54 (i.e., N_l) landmark points.

The maximum search level $D_{\max}$ imposed by the algorithm ensures that the number of landmarks is limited. This size limitation is needed for both computational efficiency and feature characterization. As will be shown in the next section, a set of new invariant features are developed, and they are based on the number of landmarks. By limiting the number of maximum landmarks, new invariant features can be represented by a vector of fixed dimension, which is critical for automatic classification using neural network learning.

In essence, the landmark set provides an approximation of contour shape features in an image. Constant ϵ can be viewed as the measure of resolution, and its value typically depends on engineering choices. If ϵ is sufficiently small and $D_{\max}$ is sufficiently large, $\Omega = X_p$. As aforementioned, the compression from X_p to Ω is important and necessary. The proposed recursive algorithm ensures that landmarks are properly spread along any (non-uniform) contour via an efficient and uniform search through a tree-like structure.

Obviously, choices of $D_{\max}$ and ϵ require tradeoffs between accuracy and computational efficiency. In general, searching depth and/or landmark resolution should be increased as complexity of the shape contour increases. For a very complicated contour, searching depth and landmark resolution can be set locally for parts of the contour. For instance, one can first apply an algorithm of water flow to determine such features as loop, loop size, and variations of gradient. Using a criterion based on these features, searching depth and landmark resolution can be selected intelligently and automatically.

4. A Vector of New Projection-Invariant Features

In many applications, 2-D images are obtained by projecting a 3-D object onto the imaging plane. In case that the depth of the object is much smaller than the distance between the object and the projection plane, shape contours in the images can be treated as ones of 2-D. For moving objects, their shape contours will change as a function of projective mappings. To classify distinct but unknown objects based on image sequences, a generic projection-invariant measure is developed in this section for distinguishing general objects so that automatic classification can be pursued in the next section by means of learning.

To this end, consider four distinct points on the shape contour in an image: p_i, p_j, p_k, and p_l. Using coordinates of any three points (say, p_i, p_j, and p_k), one can formulate the following determinant:

$$v(i, j, k) = \frac{1}{2} \begin{vmatrix} x_i & x_j & x_k \\ y_i & y_j & y_k \\ 1 & 1 & 1 \end{vmatrix}. \tag{4.1}$$

Note that $v(i, j, k) \neq 0$ unless the three points are on a straight line.

For general projective images, cross ratios of a linear shape measure are invariant to affine transform action [2, 15]. When the perspective changes, the four points in the new image become p'_i, p'_j, p'_k, and p'_l, and the relationship between p_i and p'_i is given by an affine transformation

$$p'_i = Ap_i + b,$$

where A is the rotational/scaling matrix and b is the translation vector. By employing the homogeneous representation, we have

$$\begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = \begin{bmatrix} A & b \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \triangleq T \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}.$$

Therefore, if adopted as a feature, ratio

$$\frac{v(i,j,k)}{v(i,j,l)} \tag{4.2}$$

is projectively invariant as expected since

$$v'(i,j,k) = \frac{1}{2} \begin{vmatrix} x'_i & x'_j & x'_k \\ y'_i & y'_j & y'_k \\ 1 & 1 & 1 \end{vmatrix} = \frac{|T|}{2} \begin{vmatrix} x'_i & x'_j & x'_k \\ y'_i & y'_j & y'_k \\ 1 & 1 & 1 \end{vmatrix}$$

and consequently

$$\frac{v(i,j,k)}{v(i,j,l)} = \frac{v'(i,j,k)}{v'(i,j,l)}.$$

Ratio (4.2) is both shape relevant and projection invariant and will be adopted as the seed feature. This is because determinant (4.1) is a measure corresponding to geometric area of a triangle formed by landmark points. Specifically, the triangle with vertices at p_i, p_j and p_k has the following area:

$$\mathcal{A} = \tfrac{1}{2}|(p_i - p_k) \times (p_j - p_k)|.$$

On the other hand, it follows that, since the value of a determinant is invariant under elementary operations,

$$v(i,j,k) = \frac{1}{2} \begin{vmatrix} x_k & y_k & 1 \\ x_i & y_i & 1 \\ x_j & y_j & 1 \end{vmatrix}.$$

Subtracting the first row from second and third rows and then multiplying the last column by $(-x_k)$ and $(-y_k)$ and adding the results to the first and second columns, respectively, yield

$$v(i,j,k) = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_i - x_k & y_i - y_k & 0 \\ x_j - x_k & y_i - y_k & 0 \end{vmatrix} = \frac{1}{2}(p_i - p_k) \times (p_j - p_k). \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \tag{4.3}$$

Therefore, determinant (4.1) is a signed geometric measure (related to the area). Furthermore, $v(i, j, k)$ has the property that, if p_k is changed to $(1 \pm \epsilon)p_k$ for some small constant ϵ, $v(i, j, k)$ is perturbed by the same percentage, that is, sensitivity of $v(i, j, k)$ with respect to a variation in p_i (or p_j or p_k) has a unity value, which can be simply denoted by

$$S_p^v = 1. \tag{4.4}$$

Nonetheless, ratio (4.2) is not an appropriate shape feature for unknown objects for two reasons. First, while being invariant under affine transformation, its value is too dependent upon the choices of points and thus captures too little information about the shape to be effective. Theoretically, one could overcome this problem by calculating all the values associated with combinations of all the points. However, this method of exhaustion is undesirable computationally. Second, if calculation of the ratio is selectively done for some of the points, selection of these points need to be carried out intelligently so that shape information of unknown projects can be described by a fixed feature that can be computed efficiently.

The approach taken in the Letter resolves the aforementioned difficulties by introducing a new invariant feature vector I. The vector is calculated in two steps (so as to address the aforementioned problems). First, only the landmark points will be considered for computing the invariant feature vector. It has been shown in Section 3 that, through the recursive search algorithm, information of shape contour is compressed and captured by the landmark points. Second, motivated by ratio (4.2), the following vectors of fixed dimension are calculated:

$$\Psi_i = \left[\frac{v_l(1, i+1, i+2)}{v_l(i, i+1, i+2)} \frac{v_l(2, i+1, i+2)}{v_l(i, i+1, i+2)} \quad \cdots \quad \frac{v_l(N_l - 1, i+1, i+2)}{v_l(i, i+1, i+2)} \right.$$
$$\left. \frac{v_l(N_l, i+1, i+2)}{v_l(i, i+1, i+2)} 0 \cdots 0 \right]^T, \tag{4.5}$$

where $i = 1, \ldots, N_l - 2$, $N_{\max} = 2^{D_{\max}+1} - 2$, $\Psi_I \in \Re^{N_{\max}}$, and $v_l(\cdot, \cdot, \cdot)$ are the determinants of form (4.1) for the set of landmarks. That is, given a set of landmarks $\Omega = \{q_j, \ j = 1, \ldots, N_l\}$,

$$v_l(i, j, k) = \frac{1}{2} \begin{vmatrix} q_{xi} & q_{xj} & q_{xk} \\ q_{yi} & q_{yj} & y_{yk} \\ 1 & 1 & 1 \end{vmatrix}.$$

Then, the proposed invariant feature vector is denoted by $I \in \Re^{N_{\max}}$, and its kth element is defined to be

$$I_k = \begin{cases} \sum_{i=1}^{N_l - 2} \|\Psi_i\|^2 & i = 1, \\ \frac{1}{I_1} \sum_{i=1}^{N_l - 2} \sum_{j=1}^{N_{\max}} \Psi_i(\mathrm{mod}(k + j - 1, N_{\max}))\Psi_i(j) & i = 2, \ldots, N_{\max} \end{cases} \tag{4.6}$$

where $\Psi_i(j)$ is the jth element of vector Ψ_i, and function mod(k, n) is defined to be the modulus of k by n.

Remark. It follows from (3.1) that, given the shape resolution characterized by $\epsilon > 0$, any three adjacent landmarks $(q_i,\ q_{i+1},\ q_{i+2})$ are selected not to be on a straight line. Therefore, vectors in (4.5) and feature (4.6) are well defined (i.e., there is no singularity).

The proposed feature also has the following properties:

- Vector I is invariant with respect to any affine transformation on the elements in Ω.
- Vector I is invariant with respect to the ordering in set Ω.
- Feature component I_k can be viewed as the correlation factor of vectors Ψ_i with step size k, and thus vector I is the self correlation vector of collection $\{\Psi_i\}$. Furthermore, it is straightforward to show that $|I_k| \leqslant 1$ for $k = 2, 3, \ldots, N_{\max}$ and that

$$I_1 + I_1\left(\sum_{k=2}^{N_{\max}} I_k\right) = \left\| \sum_{j=0}^{N_l - 2} \Psi_j \right\|^2.$$

- If Ω is replaced by $\Omega \cup \Omega$ (so that N_l is doubled while $N_{\max}$ is fixed and not exceeded by N_l), then the non-zero elements of Ψ_i are repeated once within the resulting new set, the corresponding value of feature I_k is increased by a constant multiplier, and the non-zero elements of vector I are repeated once within the new set as well.

The last property together with (4.4) justifies the use of landmarks in preserving shape information in I while limiting computational effort to the landmark points.

To illustrate invariance of the proposed feature vector, consider the two projections of an object in Figure 3. Landmarks are found by the recursive search algorithm, and they are shown in Figure 3. The corresponding features I_k, invariant to translation, scaling, rotation, and reference point, are calculated, verified, and then plotted in Figure 4.

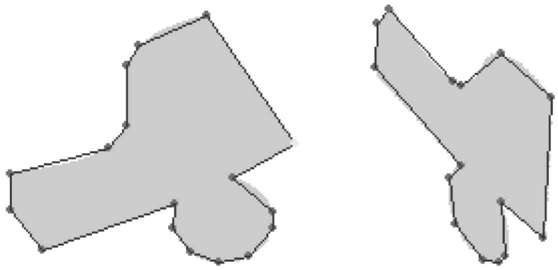

Figure 3. Landmark locations of two perspectives of an object.

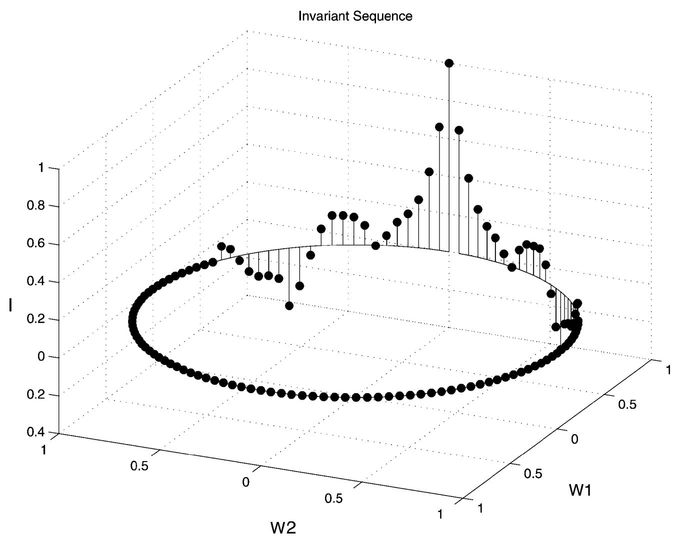

Figure 4. Circular plot of invariant sequence I_k where $w_1 = \cos(2k\pi/N_l)$ and $w_2 = \sin(2k\pi/N_l)$.

5. Automatic Classification by Unsupervised Learning

Having the capability of pattern learning and automatic recognition is one of the most essential and desirable functions of today's intelligent systems. For automatic classification, unknown patterns need to be processed, characterized, and learned so that they can be assigned into different classes or categories according to their characteristics. For online applications, a classifier must be versatile in handling pattern features, fast in online learning, and stable in classification. Adaptive resonance theory (ART) is one of the neural network classes that possess such properties. In this Letter, fuzzy ART is selected and implemented as the classifier because, unlike an ART network which operates only at discrete values, the fuzzy ART network admits inputs whose values are between 0 and 1. The basic structure and fundamental functionality of a (fuzzy) ART network is given by Figure 5. In what follows, its algorithm is illustrated.

As shown in Figure 5, the network consists of two layers: an input layer and a competitive layer. These two layers are interconnected by forward and backward weighting matrices $W_{1\to 2}$ and $W_{2\to 1}$ (both of which have non-negative entries and the initial values of their entries are typically set to be 1, and start with one competitive node, i.e. $M = 1$), respectively. When a vector of shape features (say, $x \in \Re^N$ with $0 \leqslant x_i \leqslant 1$ and for $i = 1, \ldots, N$) is presented, the state of the input layer becomes

$$X = [x_1,\ x_2, \ldots, x_N,\ 1 - x_1,\ 1 - x_2, \ldots, 1 - x_N]^T.$$

Then, a search is conducted at the competitive layer for the output node that yields the maximum response, and the network either starts to resonate or continues its

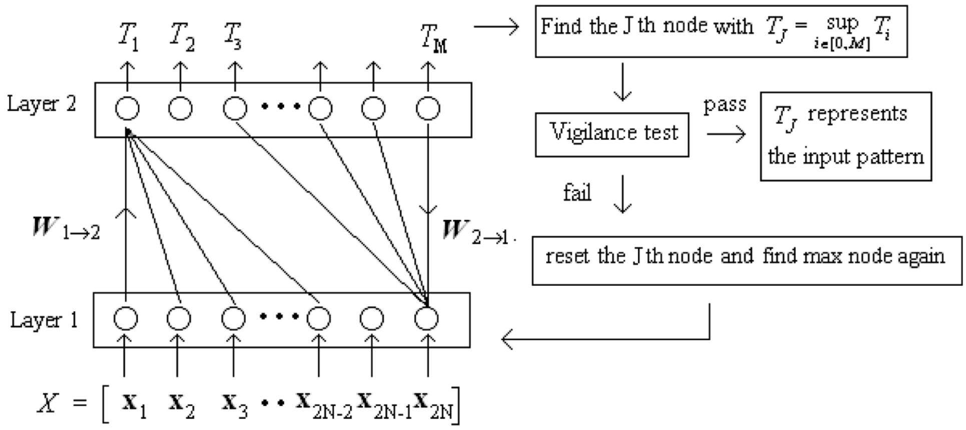

Figure 5. Fuzzy ART.

search until the best node representing the corresponding input is found. Specifically, during the search/resonance phase, the network first selects the node candidate that yields the maximum response T_J, that is,

$$T_J = \max_{W^j_{1\to 2}\in W_{1\to 2}} \frac{\| W^j_{1\to 2} \wedge X\|_1}{\beta + \| W^j_{1\to 2}\|_1}$$

where $\beta > 0$, $W^j_{1\to 2}$ is the weighting row vector from the input nodes to the jth output node, $(a \wedge b)_i = \min(a_i, b_i)$, $\|v\|_1 = \sum_{i=1}^{2N} v_i$, $W^j_{2\to 1}$ is the jth column vector of $W_{2\to 1}$ and it is set to be $W^j_{2\to 1} = [W^j_{1\to 2}]^T$. Upon identifying the Jth node, the node is validated by the so-called vigilance test, i.e., whether the inequality

$$\| W^J_{2\to 1} \wedge X\|_1 \geqslant \rho \|X\|_1 \equiv \rho N \tag{5.1}$$

holds for a vigilance constant $\rho \in [0, 1]$. If true, the node represents the closest classification of the input, its weighting will be updated by adaptation law

$$W^J_{1\to 2} = W^J_{1\to 2} \wedge X, \tag{5.2}$$

and the network is kept at to this state until the next input arrives. If false, the maximally responding node is not a valid candidate, and it will be excluded from the next round of search (by resetting the Jth node attribute so that this node is prohibited from winning in the next round of search), and the network continues its search. If there is no convergence (i.e., no valid result emerges from the search), a new node is created to represent the input. In this case, output node index M will be increased by one (that is, $M_{\text{new}} = M + 1$), $W_{2\to 1}$ adds a new column $W^{M_{\text{new}}}_{1\to 2}$, initial values of all its elements are set to be 1, and as a result $T_{M_{\text{new}}} = N/(\beta + 2N)$. Thus, in checking (5.1), a new node will be added if all the current values of T_j are less than $N/(\beta + 2N) \approx 1/2$.

In the Letter, the proposed shape recognition method is based on both the fuzzy ART network and the new invariant feature vector I in Section 4. To produce the normalized input vector, we let $N = N_{\max}$ and

$$x = [\bar{I}_1, \bar{I}_2, \ldots, \bar{I}_{N_{\max}}]^T,$$

where $\bar{I}_k \in [0, 1]$ is defined by

$$\bar{I}_1 = 1, \quad \bar{I}_k = \frac{I_k - \min I_k}{1 - \min I_k}, \quad i = 2, \ldots, N_{\max}$$

and I_k is that in equation (4.6).

In essence, the vigilance test ensures the matching condition, inequality (5.1), between the input pattern and the selected weight $W^J_{1\to 2}$, and the fuzzy ART network assigns a node (or classification) to represent the input by maximizing the corresponding response while being vigilant. Note that matching condition (5.1) and update law (5.2) form a contraction mapping. Therefore, as new input vectors come, the vigilance constant ρ can be viewed and used as the compression ratio between patterns and their corresponding nodes. Specifically, if ρ approaches one, the network trends to generate new nodes rather than update the existing ones in order to keep the match between an input pattern and its corresponding node.

In general, the adaptive resonance theory has one unique advantage over other neural networks. That is, an ART has the property of being able to learn of a new pattern without losing previously stored information. Specifically, adaptive resonance theory is developed to solve the crucial tradeoff between stability and plasticity [4]. A typical neural network tends to loss previously stored information as it starts learning a new pattern.

In the ART architecture, the solution to the problem is the use of resonance and feedback inter-connected weights that connect the output from the second layer back to the input layer, as shown in Figure 5. The feedback mechanism allows the network to either resonate or search for the best representation through different candidate nodes until an existing or a new output becomes the desired node, and that node is then considered as the closest representation among all of the nodes.

Specifically, the important feature of the ART model is its automatic switching between convergence (stable) and learning (plasticity) modes. This property is achieved by the design of the weight update law and resonance mechanism since the network updates only the chosen node while leaving all other weights intact. This prevents cross interference from happening during the network updates. Whenever a totally different pattern arrives, the resonance mechanism creates a new node to represent the pattern. As the result, these key properties make the ART network much more stable, without losing previously stored information and also automatically generate new nodes to encode new patterns when they arise.

By combining the new invariant feature vector and fuzzy ART network learning capability, the proposed method has all necessary properties and functions to achieve online and automatic shape recognition.

6. Illustrative Example

In this section, the proposed algorithm is tested for automatic classification of airplanes. The test uses a set of different geometrical perspective images of the aircraft to illustrate functionality and capability of the proposed fuzzy ART neural network based classification method. Specifically, in the simulation, we prepare a set of 16 images, $P = \{P_i, i = 1, 2, \ldots, 16\}$ for testing the network. The first four images, $P_1, \ldots, P_4$, are the pictures that appear in Figure 6 and 7, and the others are a set of different geometrical perspectives of $P_1, \ldots, P_4$ (e.g., Figure 8), denoted by $P_{4i+j} = T_i(P_j), j = 1, 2, \ldots, 4$. These images are generated by three affine transforms, $T_i(\cdot)$, $i = 1, 2, 3$. Results of the network presented by a sequence of the test images are shown in Tables I and II.

In the simulation, the fuzzy ART network and the recursive landmark parameters are set to be $\beta = 0.1$, $\rho = 0.9$, $D_{\max} = 7$, $M = 1$, and $\epsilon = 1$. The matching scores between patterns and their representing nodes and the results of the network responses are summarized in Tables I and II. It is apparent that the proposed network recognizes the types of aircrafts with high matching scores.

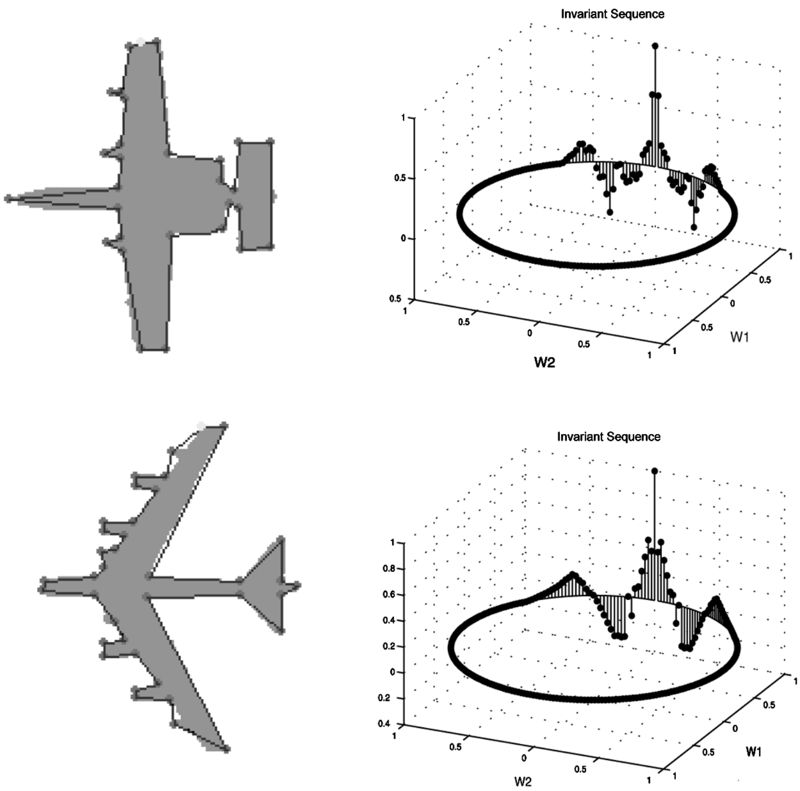

Figure 6. Planes of type-1 and type-2 with their features.

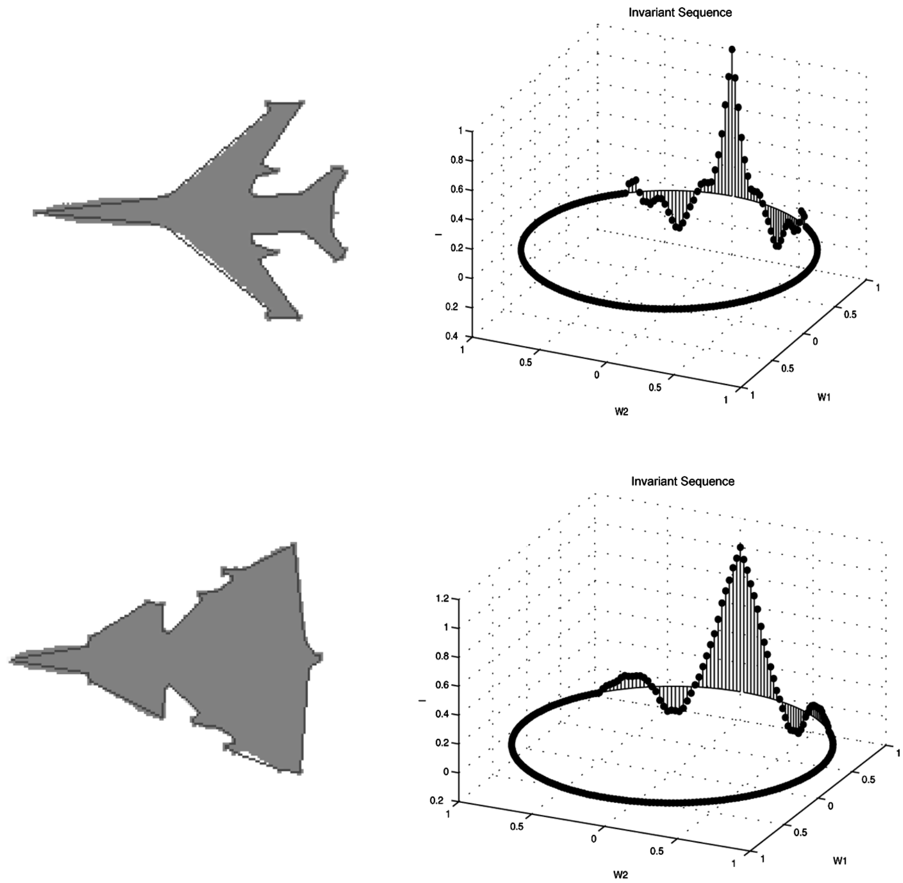

Figure 7. Planes of type-3 and type-4 with their features.

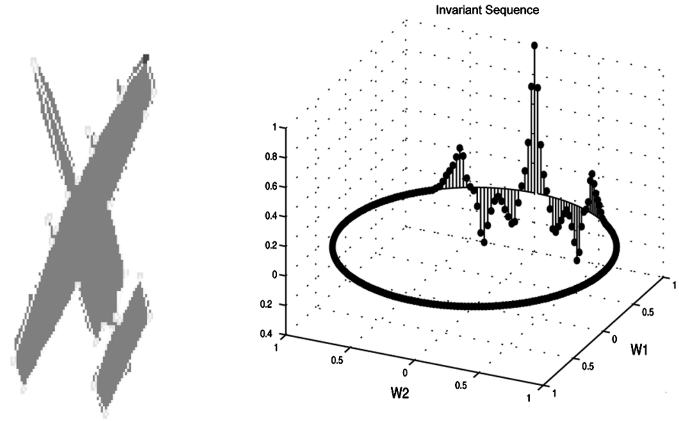

Figure 8. Testing image and its invariant features.

Table I. Network responses and matching scores of presenting images (in percentage).

Image	P_1	P_2	P_3	P_4
Network response	create node 1	create node 2	create node 3	create node 4
Matching score	100	99.9	99.9	99.9
Image	P_5	P_6	P_7	P_8
Network response	match node 1	match node 2	match node 3	match node 4
Matching score	97.8	99.5	94.8	95.8

Table II. Network response and matching scores of presenting images (in percentage).

Image	P_9	P_{10}	P_{11}	P_{12}
Network response	match node 1	match node 2	match node 3	match node 4
Matching score	97.5	98.7	93.9	95.1
Image	P_{13}	P_{14}	P_{15}	P_{16}
Network response	match node 1	match node 2	match node 3	match node 4
Matching score	96.1	97.5	93.5	96.2

7. Conclusion

A method of shape recognition using landmark invariant features and a fuzzy ART network is presented. It is shown to possess such unique properties as learning and recognizing object shape online with a combination of invariant features against weak perspective distortion and stable learning capability of fuzzy ART network. The proposed method has the potential of enabling automatic shape learning and recognition in many industry applications. Illustrative simulations have been carried out to demonstrate the effectiveness of the proposed method, and a summary of results is included.

Acknowledgement

This research is supported in part by Microtronic Inc. We would like to thank the editor and reviewers for their help.

References

1. Antoine, J.: Shape characterization with the wavelet transform, *Signal Processing*, **62**(3) (1997), 265–290.

2. Barrett, E. B., Brill, M. H., Haag, N. N. and Payton, P. M.: Some invariant linear methods in photogrammetry and model-matching, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 122–128, 1992.
3. Barrett, E. B. and Payton, P. M.: Geometric invariant for rational polynomial cameras, *Proceedings of Applied Imagery Pattern Recognition Workshop*, **29** (2000), 223–234.
4. Carpenter, G. A., Grossberg, S., Markuzon, N. and Reynolds, J. H.: Fuzzy ARTMAP: A neural network architecture for incremental supervised learning for analog multidimension maps, *IEEE Transactions on Neural Networks*, **5** (1992), 698–713.
5. Carpenter, G. A., Grossberg, S. and Rosen, D. B.: Fast stable learning and catergorization of analog patterns by an adaptive resonance system, *Neural Networks*, **4** (1991), 759–771.
6. Fischler, M. A., Wolf, H. C.: Locating perceptually salient points on planar curves, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**(2) (1994), 113–129.
7. Freeman, H.: On the encoding of arbitrary geometric configurations, *IRE Transactions on Electronic Computers*, **10** (1961), 260–268.
8. Granlund, G.: Fourier preprocessing for hand print character recognition, *IEEE Transactions on Computers*, **21**(2) (1972), 195–201.
9. Kim, S. D., Lee, J. H. and Kim, J. K.: A new chain-coding algorithm for binary images using run-length codes, *Computer Vision, Graphics and Image Processing*, **41** (1988), 114–128.
10. Luo, D., Macleod, J. E. S., Leng, X. and Smart, P.: Automatic orientation analysis of particles of soil microstructures, *Geotechnique*, **42** (1992), 97–107.
11. Morel, J. M. and Soliminii, S.: *Variational Methods in Image Segmentation*. Boston: Birkhauser, 1995.
12. Mundy J. L. and Zisserman, A.: *Geometric Invariance in Computer Vision*. Cambridge, Mass.: MIT Press, 1992.
13. Qu, Z.: *Robust Tracking Control of Robotic Manipulators*. New York: IEEE, 1996.
14. Ramer, U.: An iterative procedure for the polygonal approximation of plane curves, *Computer Graphics and Image Processing*, **1** (1972), 244–256.
15. Reiss, T.: *Recognizing Planar Objects Using Invariant Image Features*. New York: Springer-Verlag, 1993.
16. Sarkar, D.: A simple algorithm for detection of significant vertices for polygonal-approximation of chain-codeed curves, *Pattern Recognition Letters*, **14** (1993), 959–964.
17. Sklansky, J.: Measuring concavity on a rectangular mosaic, *IEEE Transactions on Computers*, **21** (1972), 1355–1364.
18. Zahn, C. T. and Roskies, R. Z.: Fourier descripters for plane closed curves, *IEEE Transactions on Computers*, **21**(2) (1972), 269–281.