

Pattern Analysis by Active Learning Method Classifier

Mohsen Firouzi^{*1,2}, Saeed Bagheri Shouraki² and Iman Esmaili Paeen Afrakoti²

¹Graduate School of Systemic Neurosciences, Ludwig Maximilian University of Munich, Munich, Germany

²Research Group of Brain Simulation and Cognitive Science, Artificial Creatures Lab, Electrical Engineering School, Sharif University of Technology, Azadi Avenue, Tehran, 11365-9517, Iran

Abstract. Active Learning Method (ALM) is a powerful fuzzy soft computing tool, developed originally in order to promote an engineering realization of human brain. This algorithm, as a macro-level brain imitation, has been inspired by some behavioral specifications of human brain and active learning ability. ALM is an adaptive recursive fuzzy learning algorithm, in which a complex Multi Input, Multi Output system can be represented as a fuzzy combination of several Single-Input, Single-Output systems. SISO systems as associative layer of algorithm capture partial spatial knowledge of sample data space, and enable a granular knowledge resolution tuning mechanism through the learning process. The knowledge in each sub-system and its effectiveness in the whole system would be extracted by Ink Drop Spread in brief IDS operator and consolidated using a Fuzzy Rule Base (FRB), in order to acquire expert knowledge. In this paper we investigate ALM as a conspicuous classifier in different types of classification problems. Also, a new ALM architecture to actively analyze ill-balanced image patterns is proposed. Different types of data sets are used as a benchmark, including a remote sensing image classification problem, to evaluate the ALM Classifier (ALMC). With active pattern generation ability and knowledge resolution tuning, ALMC has been distinguished from many conventional classification tools especially for complex structures and image patterns analysis. This work demonstrates that ALMC is a good noise robust and active classifier, which is adaptively adjusted through structural evolution and pattern evaluation mechanism. These remarkable capabilities, along with its straightforward learning process, make ALMC as a convenient soft computing tool to use in different types of low dimensional pattern recognition problems.

Keywords: Active Learning Method, Adaptive Neuro-Fuzzy Classifier, Remote Sensing Image Classification, Radial Base Function Network, Support Vector Machine

1. Introduction

The brain machine is often studied through research themes with different disciplines in order to exploit human intelligence. The human brain has always been studied in two major levels of abstraction, micro-level and macro-level. The micro-level studies started from neurophysiological findings of interactions among different types of neurons in vivo and correspondingly led to connectionism paradigms, such as ANNs in Artificial Intelligence. Macro-Level brain studies, on the other hand, are originated from psychological researches with the objective to understand behavioral features of human-environment interactions. Symbolism in AI, in which human-like in-

telligent machine models surrounding phenomena by creating formal symbols and try to acquire knowledge by manipulating these qualitative objects, arises from this viewpoint of brain studies. Rule-based systems such as Expert System and FRB are good examples of this paradigm [1].

Despite many debates, today's main secrets of brain machine have been dramatically uncovered in micro-level and macro-level attitudes. One of the specific abilities of human brain is analysis of visual, auditorial and odor sensory patterns. This information is captured and conveyed by nervous signals and analyzed for perception and recognition process. Human brain often tries to categorize its sensory experiences and explore to find their association, and

* Corresponding author, Tel: +98-21-66165984, E-mail: mfirouzi@alum.sharif.edu

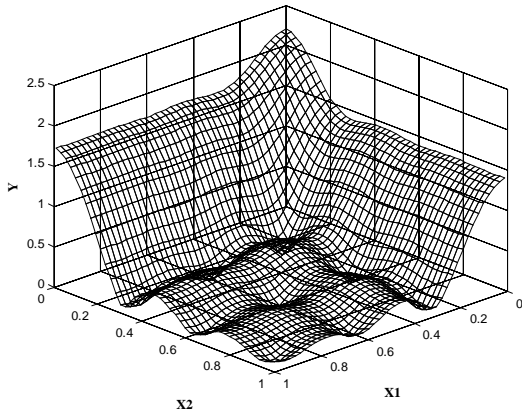


Fig.1 A simple two-input single-output system

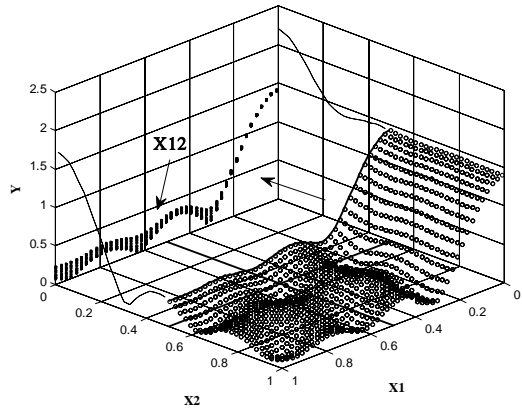


Fig.2 IDS unit X12 and the part of system characteristic surface of potential projected data points.

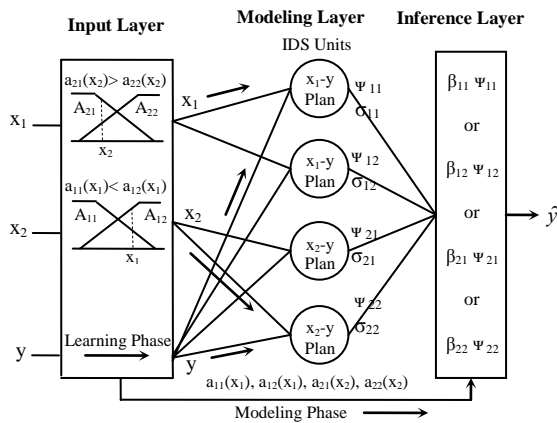


Fig.3 General Structure of a double-input single-output ALM with 4 IDS units and characterized by three layers

not just use a supervisory teacher. It is not exaggerative to address active learning as one of the most important characteristic of human learning; the active

cycle of the learning consists of the evaluation of input patterns in accordance with environmental feedback and stored prior knowledge, and strengthening (rewarding) the most valuable patterns and eliminating destructive patterns (penalty). This cycle of learning is conducted in absence of a supervisor using trial and error. From a different point of view, this learning mechanism acts like a reinforcement learning in which the human is the actor and the critic at same time.

ALM is a macro-level realization of brain with a novel fuzzy attitude inspired by two significant features of human information processing: active learning and brain qualitative modular processing with no complex mathematical computations [2]. Brain composes distributed modular structures and different areas processing different types of sensory patterns [3]. There are complex interconnections and interactions among these modular sub-systems. For example, the taste is strongly influenced by its odor [4]. Correspondingly, psychological evidences have demonstrated that human handles real world complex systems as a combination of simpler and more comprehensible aspects of entire system [5], This helps to consume less energy to acquire information, and allows us to discover each input-output correlation and their influence in the whole system without complex mathematical processing. This concept is known as reductionism in philosophy of mind, in which a complex phenomenon is represented as a combination of partial simpler phenomena or as a superposition of sub-causes effects [6].

ALM has been developed by Shouraki to advance a new computational paradigm closely mimicking human information processing. This paradigm avoids complex mathematical computations and definitions, in contrast to the most conventional soft computer tools [7]. Distinguished features of this soft computer make it suitable to use in pattern analysis applications. In this work, we assess prominent capabilities of ALMC in comparison to some of the traditional classifiers. Different types of standard benchmarks are used, including Iris data set, a complex helical structure, Sugeno-Yasukawa function approximation and a remote sensing image classification which is performed by a new ALMC architecture for active pattern analysis.

In the next section, ALM is elaborated clearly. In section III standard benchmarks are analyzed. In section IV active analysis of a remote sensing image as a

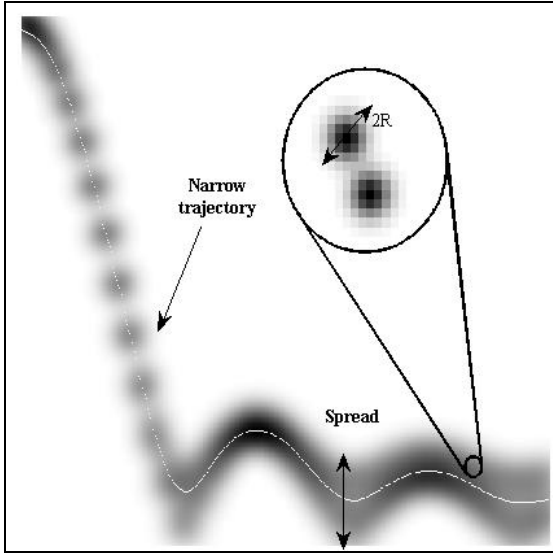


Fig.4 IDS unit X_{12} with 256×256 resolution grid plane, operated by IDS operator with $R=9$. White path: extracted narrow trajectory and double side arrow: Spread value.

realistic application is investigated and finally conclusion and remarks are presented in section V.

2. Active Learning Method

ALM is an adaptive recursive fuzzy inference system expressing a MIMO system as a fuzzy combination of simpler SISO systems. Each SISO system is described as a 2-D x_i - y grid plane called IDS unit. These units, as the processing layers of ALM, consist of (x_i, y) projected data points belonging to specific segmented intervals of other input variables domain. For instance, in Fig.1 a simple two-input single-output system is shown. If each input domain is segmented to two regular intervals, there is overall four IDS units which capture related data points: $X_{11} = \{(x_1, y) | x_2 \in [0, 0.5]\}$, $X_{12} = \{(x_1, y) | x_2 \in [0.5, 1]\}$, $X_{21} = \{(x_2, y) | x_1 \in [0, 0.5]\}$ and $X_{22} = \{(x_2, y) | x_1 \in [0.5, 1]\}$. Fig.2 shows X_{12} IDS unit and the part of system characteristic surface including potential projected data points on X_{12} . Intervals can be segmented to smaller sub-intervals that increase overall IDS unit numbers. This structural evolution acts like a granular knowledge resolution tuning. It means that IDS units related to smaller intervals capture higher resolution of spatial content, in spite of their lower chances in data capturing, causing sparser data plane. It is depicted in section III that in cases with large learning data set

size, partitioning number should be large enough to achieve more accurate modeling and vice versa. On the other hand, X_{12} in Fig.2 illustrates that in some intervals of x_1 , data points are thinly scattered. So that correlation between output and x_1 in the intervals with large dispersal area is less than intervals with small spread. This difference arises from the fact that in larger dispersal areas of x_1 there is more possible values to choose as SISO output characteristic. So this feature indicates how much output value of SISO depends on the related value of x_1 .

A simple architecture of ALM for a double-input, single-output system with four IDS units is shown in Fig.3. ALM has three layers. Input layer, in which input patterns are, distributed in IDS units. Modeling layer, the associative layer of ALM operates as most important part of ALM which extracts SISO sub-systems' characteristic functions, hereafter Narrow path and its effectiveness in entire system or Spread value. Since IDS units, or SISO sub-systems, capture specific partial spatial content, these two features are called partial knowledge in pervious works [8]. In learning phase projected data points are fed by a fuzzy curve fitter operator (IDS operator) to extract narrow and spread values. In the modeling phase narrow and spread are delivered to inference layer to consolidate partial knowledge and make top level decision. In Fig.3 ψ_{ij} and σ_{ij} parameters stand for narrow and spread values respectively. In the next section, mathematical outlook of IDS is described.

2.1. Ink Drop Spread Operator

IDS operator is a fuzzy curve fitting technique operating on scattered data plane of IDS units to interpolate narrow trajectory and extract spread values. This knowledge extraction engine is considerably similar to the fuzzy patch proposed by Bart Kosko [9]. It is like a Gaussian Ink Drop with radius R spreading around each projected data points. As a fuzzy technique, Ink Drop acts like a radial base membership function around crisp data points, and provides none-precise representation of the experimental data. Fig.4 shows X_{12} IDS unit which is fed by Ink Drops. Darker areas means bigger three-dimensional membership value for related (x_i, y) point. Also for the points whose Euclidian distance is less than R , ink Drops overlap together and ink intensities become more concentrated (*knowledge overlapping property*). Depending on different constraints in different applications, mathematical implementa-

tion of narrow and spread could be different [2], [5], [8]. But as a general aspect, the value of narrow path for each x_i is being calculated by Center of Gravity Defuzzification (COGD) over fuzzy-like (ink intensity) weighted (x_i, y_k) points (ψ in Fig.3). As discussed before, Spread is a parameter showing the output's certainty around narrow path and indicates effectiveness degree of corresponding input in overall system output. So, smaller spread value shows bigger effectiveness and vice versa.

In this work Mathematical form of Ink Drop, narrow and spread implementation is as following equations. It is worth to mention that hereafter all input and output values have been normalized before training. Let's suppose (p_1, p_2) is the point of Ink Drop (center of Gaussian membership function) on X - Y plane and $d(x, y)$ denotes the darkness level at (x, y) so:

$$\Delta d(x, y) = e^{-\frac{(x-p_1)^2}{0.22 \times R^2}} \times e^{-\frac{(y-p_2)^2}{0.22 \times R^2}}, \quad (1)$$

$$\forall (x, y), \sqrt{(x-p_1)^2 + (y-p_2)^2} \leq R$$

Narrow path and spread are described as (2) and (3) respectively:

$$\psi(x_i) = \frac{\sum_{j=1}^n I_{ij} \times y_{ij}}{\sum_{j=1}^n I_{ij}}, \quad 1 \leq i, j \leq n \quad (2)$$

Where n is IDS grid plane resolution, I_{ij} denotes as Ink intensity or fuzzy-like membership degree for each y_j values corresponding to x_i .

$$\sigma(x_i) = \max \{y_{ij} \mid d(x_i, y_{ij}) > 0\} - \min \{y_{ij} \mid d(x_i, y_{ij}) > 0\} \quad (3)$$

If $\exists x, \{y \mid d(x, y) > 0\} = \emptyset$, there is no knowledge and no experiences about the x points. Therefore, the most conservative value for narrow and maximum value for spread must be chosen, as below:

$$\psi(x) = \frac{y_{\max} - y_{\min}}{2}, \quad \sigma(x) = y_{\max} - y_{\min} \quad (4)$$

In Fig.4 extracted narrow trajectory and spread value in X_{12} unit is shown by a white path and double side arrow respectively. It is noticeable that the size of Ink Drop pattern influences the model accuracy, depending on the training set size. Tuning of Ink Drop size and spatial partitioning of input domains

enable a resolution tuning mechanism for ALM [8]. In sparser data planes, it is necessary to apply a large Ink nevertheless consenting of the spread value increasing. In fact IDS is an inference engine of ALM that identifies operands for inference rules.

2.2. Inference layer

In modeling phase, fuzzy inference layer uses narrow and spread values to unify partial knowledge and extract entire system characteristic. For each IDS unit, there is a fuzzy rule. In the case of N -input variable with m_i partitions, the number of combination rules and IDS units related to i th input, denoted by l_i , and the total number of rules, denoted by L , is as follows:

$$l_i = \prod_{k=1, k \neq i}^N m_k \quad (5)$$

$$L = \sum_{i=1}^N l_i = \sum_{i=1}^N \prod_{k=1, k \neq i}^N m_k \quad (6)$$

Also the k th rule of the i th input variable, R_{ik} ($k = 1, 2, \dots, l_i$) can be described as below:

$$R_{ik} : \text{if } x_1 \in A_{j_1}^1 \wedge x_2 \in A_{j_2}^2 \wedge \dots \wedge x_{i-1} \in A_{j_{i-1}}^{i-1} \wedge x_{i+1} \in A_{j_{i+1}}^{i+1} \wedge \dots \wedge x_N \in A_{j_N}^N \text{ then } Y = \psi_{ik}(x_i) \quad (7)$$

Where $A_{j_s}^s$ is j_s -th segment of S -th input and $1 \leq S \leq N$, $S \neq i$. As discussed in this section, IDS unit X_{ik} includes projected data points belonging to specific partitions of other variables. In the modeling phase, these sub-domains have been described fuzzy intervals. So rule R_{ik} could be activated when all antecedent terms of $A_{j_s}^s$ has non-zero membership degree in (7). Then X_{ik} - Y plane would be activated by specific truth degree and its partial knowledge including narrow and spread will take part in top level decision making process. Consequently, the overall output is obtained by Min-COGD composition through fuzzy rules, as follow:

$$Y \text{ is } \beta_{11} \psi_{11} \text{ or } \dots \text{ or } \beta_{ik} \psi_{ik} \text{ or } \dots \text{ or } \beta_{Nl_N} \psi_{Nl_N} \quad (8)$$

In which *or* is union operation (S-Norm) and β_{ik} is normalized term of spread reverse value. The truth degree of R_{ik} described by (9):

$$\beta_{ik} = \frac{\alpha_{ik}\gamma_{ik}}{\sum_{p=1}^N \sum_{q=1}^{l_p} \alpha_{pq}\gamma_{pq}}, \alpha_{ik} = \log\left(\frac{1}{\sigma_{ik}}\right) \quad (9)$$

In equation (9), α_{ik} is the normalized reverse value of Spread for IDS unit X_{ik} , and γ_{ik} stands for R_{ik} truth degree. The logarithmic function as a depression mapping has been used to smooth sharpness of spread reverse value and general modeling surface.

Because of matrix-like architecture of IDS, memory cost is a drastic challenge in ALM [2], [5]. An arithmetical form of IDS in a pipelined architecture with similar functionality and better storage performance, called PRIDS, has been developed by Firouzi *et.al* [10]. In contrast to IDS, PRIDS is an iterative operation. But its pipelined form of arithmetical calculation makes it a considerably fast method for hardware applications and free from matrix-like storage. Also recently, Merrikh-Bayat *et.al* proposed an analog hardware solution based on Memristor Crossbar that is more compatible with real time and non-exact nature of IDS, in contrast to the digital approach [11].

3. Active Learning Method Classifier

From the learning point of view, ALM is a supervised online learning algorithm with conspicuous specifications, such as noise immunity and rapid straightforward learning process. As opposed to most of the classical supervised soft computers, such as SVM as a kernel base classifier, ANNs and Adaptive Neuro-Fuzzy Classifier (ANFC), ALM is free from iterative mathematical computations to achieve learning convergence. Therefore, it could provide a classifier with distinguished abilities. In this section, these abilities are inspected and compared with the traditional classifiers using three standard benchmarks.

3.1. Iris Data Set

Iris data set is a famous simple classification benchmark to evaluate new algorithms. It contains 150 four dimensional data of three types lily flower, in which two classes are not linearly separable. In the learning phase 75 random data patterns are chosen and other 75 samples is used for evaluation process and does not take part in learning. All input and output patterns are normalized between zero and one.

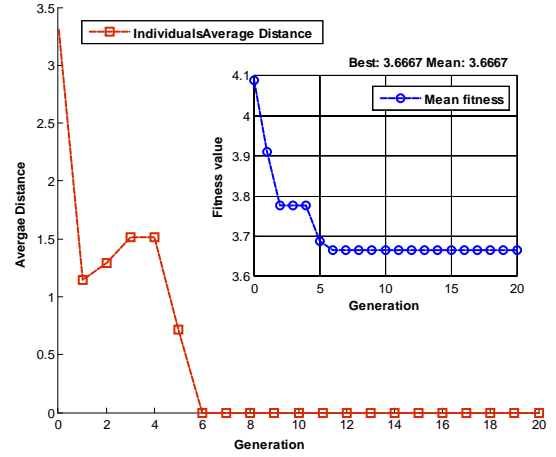


Fig.5 Evolution diagrams of (big diagram): Individual average distance for each population generation, (Small): Mean Fitness value.

Also, ten randomly selected training and validation sets are used to evaluate classifiers.

In this problem, ALMC is architected as four-input, single-output system. Resolution of IDS grid planes is set as 256×256 . But since different forms of partitioning and ink drop pattern generate different classification rates, in order to acquire most efficient architecture with less redundancy and best performance, an evolutionary optimization solution is proposed in which mean classification rate over all random data batches is examined by evolution of ink radius and input partition numbers. GA is a heuristic evolutionary computation tool that is theatrically eliminates computational cost of massive optimization problems. In this work a GA machine with population size of 6, Gaussian mutation function, scattered crossover function and uniform stochastic selection function has been used. Fitness function is the mean classification error over all 20 random sets. This function has 5 input variables: partition numbers of each input variable and ink pattern radius. Evolution of mean fitness value and mean individual distances over 20 generation has been described in Fig.5. Algorithm is saturated with the best fitness value of 3.66% after 6 generations. Then best individual is selected for ink pattern size and partition numbers. Consequently, ALMC is set up by 1504 IDS units in accordance with 6, 8, 8, and 7 partitions for each input variable and Ink radius of 10 (4% of plane resolution). Also, because of the single output architecture, pattern coding is performed like an ordinary binary output coding in FFNs [12]. So the output domain is divided into three

Table.1 Mean classification rate of validation and training data sets over 10 random sets of Iris

Iris Data Set								
Classification Method	Training Set				Validation Set			
	Classification Rate (%)	Standard Deviation	Max Rate (%)	Min Rate (%)	Classification Rate (%)	Standard Deviation	Max Rate (%)	Min Rate (%)
<i>ALMC</i>	98.27	1.18	100	96.00	94.40	1.66	97.33	92.00
<i>RBFN</i>	99.60	0.65	100	98.66	94.66	1.78	97.33	90.66
<i>FFNN (LM⁺)</i>	96.13	1.78	97.33	93.33	89.33 %	6.28	97.33	78.67
<i>SVM (RBF[*])</i>	98.98	0.84	100	97.78	96.13%	1.13	97.78	94.22

⁺ Levenberg-Marquardt Back-Propagation learning algorithm

^{*} Radial Base Function Kernel

intervals as output label values and two boundary intervals as unknown values.

The results of ALMC mean classification rate over ten random training and validation sets in comparison to Feed-Forward Neural Network, Radial Base Function Neural Network and Support Vector Machine are listed in Table. I. FFNN is a four-input, three-output network with two hidden layers of 5 and 35 sigmoid neurons. Also, Levenberg-Marquardt Back-Propagation learning algorithm as the most effective BPL algorithm has been used with learning rate of 0.001.

RBFN is a four-input, three-output fixed center strategy network with Gaussian receptive fields. Fixed center network is an expanded form of generalized RBF network in which single basis function has been centered on a single learning pattern [13]. Receptive field spread value is set at 0.04, to be consistent with the ink drop radius that is set at 4% of input variables resolution. There is sort of resemblance between radial base function receptive fields in RBFN and Ink Drop in ALM. When an Ink Drop is exerted on a crisp pattern, a three dimensional radial base membership function (IDS), with specific radius and center of input data, captures spatial neighborhood of the pattern. Moreover, experiments show that the effect of single Ink Drop in the overall system is similar to a non-symmetrical receptive filed [14]. Similarly, in RBFN fixed point network, a single pattern generates a single Gaussian receptive field that is about to catch spatial content around data point. In accordance with the SVM classifier, Gaussian function with 0.02 value of sigma is chosen as the kernel function. Also, Sequential Minimal Optimization (SMO) method is used to find the separating hyperplane.

Table I. shows mean classification rate of ALMC compared to RBFN, FFNN and SVM. As you see, ALMC classification rate is comparable with RBFN and SVM. In contrast to FFNNs, ALMC needs no

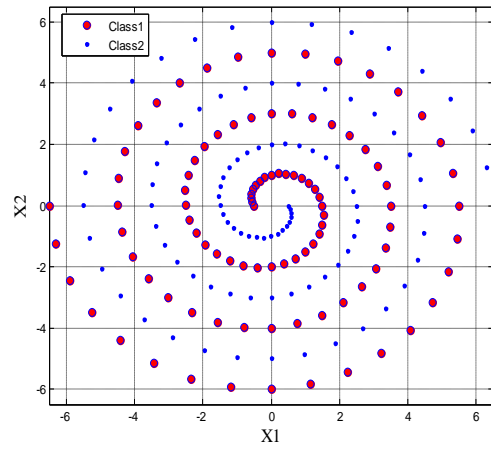


Fig.6 Data points of typical two-spiral classes with three twirls in Cartesian diagram

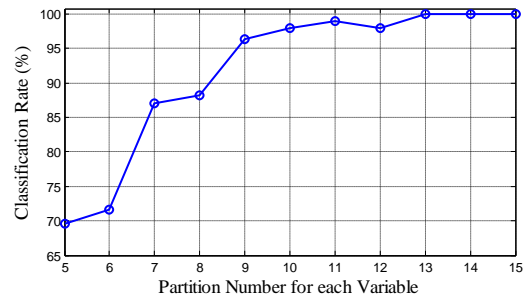
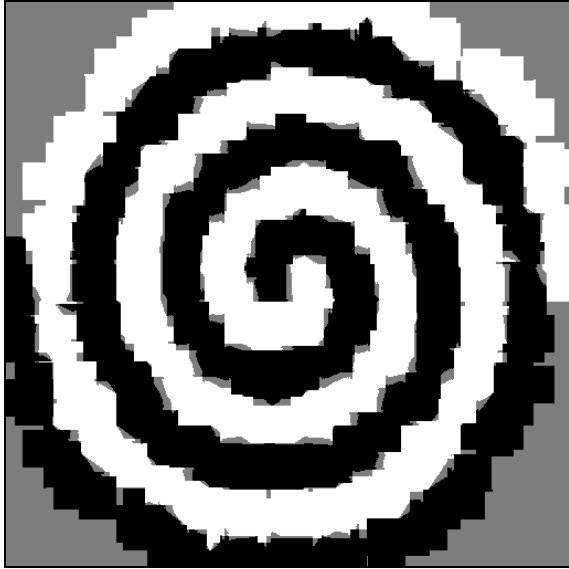
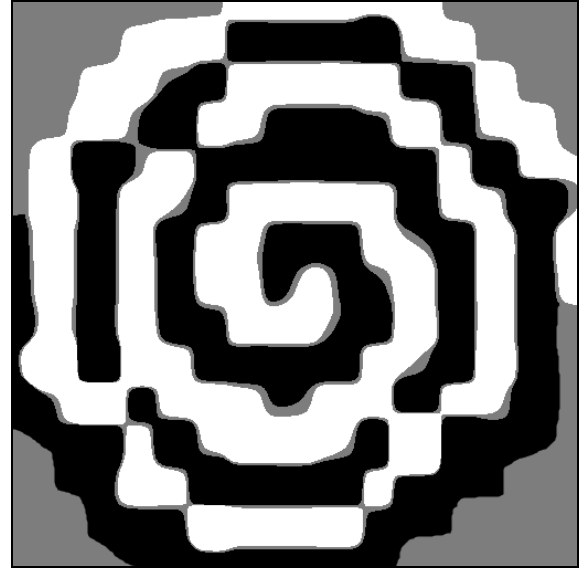


Fig.7 Classification rate through ALMC structural evolution in two-spiral problem

random parameter initializing in the training phase. This advantage provides more stability in the classification phase and more tolerance for the parameter uncertainty. This is shown in Table I, where ALMC has less standard deviation over ten random sets compared to FFNN over both validation and training sets.



(a) ALMC



(b) ANFC

Fig.8 The 512×512 pixel image of classified two-spiral structures using:

(a): two input, single output ALMC with 15×15 triangular-shaped fuzzy partitioning (225 rules), Ink Radius 9 and 256×256 IDS grid planes.

(b): ANFC with 15×15 bell-shaped membership functions (225 rules) and hybrid learning (least square-BPL) with 200 epochs.

Black: Class 1, White: Class 2, Gray: Unknown (Binary output Coding)

3.2. Classification of Complex Helical Structure

One of the well-known intricate classification benchmarks is the two-spiral problem with two intertwined helical structures [15]. Each spiral structure is comprised of 97 data points. There are 194 points in total, located on the polar plane according to (10):

$$r = p(\theta + 2\pi m) + r_0 \quad (10)$$

Where r and θ is radius and angel in polar plane and p is the parameter that determines the size of the spiral. r_0 and n represent the start point radius and number of spiral rotation respectively. In this work typical values of $p = 1/\pi$, $r_0 = 0.5$ and $n=3$ are used. In Fig.6, data points of typical two-spiral structure with three twirls in Cartesian diagram are shown.

Traditional ANN classifiers such as Multi-layer Perceptron with standard BP learning are trapped in local minimums and cannot perform a stable classification in this problem [16]. Therefore, it needs a network to learn a highly nonlinear separation. It is possible to completely classify this complex structure only by sophisticated preprocessing on patterns [15] or using FFNNs with refined architecture and learning algorithm. For instance Cascade-Correlation ar-

chitecture has better performance in this problem, with faster convergence in comparison to typical FFNNs [17]. Ikuta *et.al* developed a feed-forward network with one additional layer of chaotic oscillatory neurons to classify two spirals [16]. In some methods, geometrical nature of spirals is considered as angel and radius imposed to a knowledge-based neural network. This method inhibits better performance and fast convergence [18]. Liang proposed a Monte Carlo learning method for MLP and achieved successful classification of two spirals using this network with 30 hidden neurons. Liang's network needs about 10^7 iterations to prosperously separate two structures [19].

All mentioned works emphasize natural complexity of this problem. Therefore, most successful methodologies suffer from massive computations and complex learning process and architectures to achieve full classification. So the supervised algorithms that can perform a complete classification of two-spiral without any preprocessing and refinements, is the superior classification solution. Adaptive Neuro Fuzzy Classifier is the successful supervised solution for this intricate problem [20]. ANFC is a kind of Takagi-Sugeno-Kang (TSK) fuzzy inference system, in which fuzzy membership functions could be adaptively adjusted using BPL. Despite its iterative learning process, ANFC is significantly

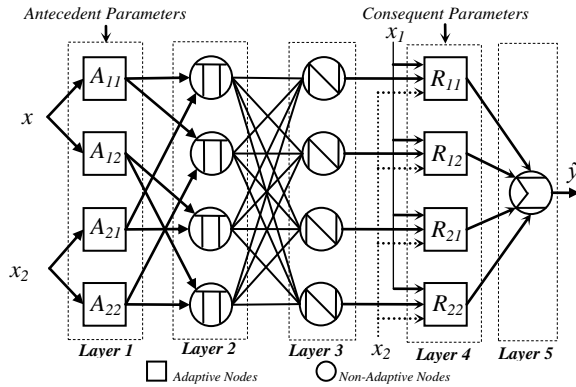


Fig.9 General Structure of ANFC with 2 inputs-single output and 2×2 fuzzy membership functions (4 Rules)

faster than neural classifiers. Fundamentally, this ability arises from considering spatial feature of patterns through adaptive adjusted fuzzy membership functions. Similar to ANFC, ALMC has made possible spatial knowledge resolution tuning through pattern space segmentation and adjusting the size of Ink Drop. This capability is a top-down structural evolution that allows granular knowledge refining from conservative view to more comprehensive insight. Also in contrast to ANFC and neural classifiers, ALMC uses a pattern-based learning mechanism free from iterative computational process.

So to achieve a full classification rate, ALMC architecture is established from a low resolution to higher ones, searching for complete classification rate. Fig.7 presents the evolution of partition numbers to meet complete classification. Due to symmetrical geometry of spiral patterns, the partition number for each variable is set to equal values. As shown in Fig.7, the minimum partition number for ALMC to obtain 100% classification is 13, generating 169 rules. The radius of Ink Drop is 9 and grid planes are 256×256 . It is noticeable that to achieve full classification by ANFC, at least 13 fuzzy membership functions are necessary for each input variable as like as ALMC (169 overall fuzzy inference rules) [20].

Besides providing a highly nonlinear separable benchmark, two-spiral is also useful in visual analysis of two twined objects. The output of a 512×512 pixels image over Cartesian plane of Fig.6, classified by ALMC and ANFC, is shown in Fig.8. ALMC has 15×15 partitions, with the ink Drop size and grid plane resolution set as 9 and 256×256 . Similarly, ANFC consists of 225 rules made by 15×15 bell-shaped membership functions. Membership functions have been adjusted using hybrid learning algorithm (Least Square Error-Gradient Descent BPL) with 200

epochs. As a consequence of visual inspection of Fig.8, both classifiers clearly achieved full classification, but ALMC results in a more obvious twined outcome in comparison with ANFC.

In this work, ALMC has been set up as a two input, single output system in which the output values above 0.6 are marked as class 1 (Black color in Fig.8) and values less than 0.4 are labeled as class 2 (White color in Fig.8). The values between 0.4 and 0.6 are classified as unknown output (Gray color in Fig.8). This coding scheme is known as binary output coding in FFNs [12].

There is no standard time criterion for this problem in the literature. But mean learning and classification run time per each pattern in ALMC with above parameters is $12.77ms$ and $1.46ms$ respectively. Mean learning and classification time for ANFC is $2.28s$ and $0.95ms$. Due to similar fuzzy modeling phases in ANFC and ALMC, classification times for the two classifiers are comparable. However, since the main part of the learning process in ANFC is performed as BPL iterative process, the learning run time in ANFC is two orders of magnitude longer. In the next subsection a comparative description of ALMC and ANFC is elaborated.

3.2.1. ALMC versus ANFC

Since both ALMC and ANFC are addressed as adaptive fuzzy inference systems, their similarities and advancements should be explained. In fact, ANFC (ANFIS Classifier Machine) is a TSK fuzzy inference system with adaptive antecedent and consequent parts. This adaptive form of TSK is known as a major step forward in machine learning revealing a new hybrid paradigm, eliminating the lack of systematic learning in fuzzy systems. As shown in Fig.9, ANFC includes five general layers. Square nodes depict adaptive nodes in which parameters are adapted during the learning process to meet target output. Circle nodes stand for non-adaptive processing units. Layer1 is the fuzzification unit and consists of the antecedent part parameters. These parameters describe the form of membership functions for fuzzy sets. In Layer2, there is a fuzzy rule for each single node performed by a T-Norm operation such as Min, to specify truth degree of corresponding rule. Layer3 normalizes truth degrees and Layer4 is TSK form consequent part by which output is described as a parameterized linear combination of input variables. Due to the linearity of consequent part, its parameters could be identified by linear optimization algorithms such as LSE, ensuring a faster

convergence. But premise part parameters should be adjusted through gradient descent and iterative process of error back-propagation. Finally Layer5 carries out COGD to model output. ANFC provided a new hybrid paradigm to manipulate oral form of knowledge. But human mind information processing is handled by qualitative knowledge and never involves exact mathematical definitions [2]. In addition, human brain often analyzes information in the form of image so that events are visualized by pattern-based memory [5].

ALMC presents a novel pattern-like learning paradigm for identification of consequent terms and membership function establishment, with no repetition of error propagation. Adjustment of membership functions is performed implicitly so as they have been taken part by their importance degree (spread values) in the final Defuzzification process. In other words, instead of antecedent part, consequent part of inference system includes two parameters. In (8) and (9) β_{ik} as weight values of Defuzzification, is a normalized term of R_{ik} rule's truth degree and its importance degree (σ_{ik}). So by imposing a new leaning pattern into IDS planes or equivalently dropping new Ink Drops, narrow path (ψ_{ik}), as explicit consequent term of related fuzzy rule (see (7)), and spread value, as its partial effectiveness, would be updated. This update is conveyed as a local operation. It means all ψ and σ parameters do not need to be adjusted by single pattern learning, and just those whose IDS unit captures new patterns should be updated by new Ink Drops. This benefit arises from top-down view to MIMO system as a combination of SISO systems in ALM.

However ALMC is considerably fast and a pattern-like learning mechanism is provided by non-exact IDS operator. But it needs larger size for memory space in comparison to ANFC. This challenge definitely restricts the applications of ALMC in high dimensional space. Knowledge in ANFC is stored through adjusted parameters. So for online learning of a new pattern, it needs no information about previous patterns, and it is independent from order of patterns. But to update IDS units by a new pattern, it is necessary to store all previous ink intensities through IDS matrix-like memory. Firouzi *et.al* develops a novel biologically plausible hybrid model for ALM using sparse temporal coding on RBF spiking neurons [8]. This model, called Spike-IDS, is a translated form of Ink space into spike time coding, by which narrow and spread could be extracted and stored through spike time dependent weighting distribution of Spike Response Model (SRM) neurons.

This form of knowledge representation for inference layer enables us to use ALMC for classification with higher dimension [14].

3.3. Sugeno-Yasukawa function approximation

Generally in modeling tools *e.g.* ANNs, fuzzy TSK, ANFIS, to obtain better noise immunity and uncertainty tolerance, it is necessary to generate extra noisy data patterns in learning phase and iterate additional training process which causes run time cost. But in ALMC there is natural noise robustness ability that is emerged from IDS operator and calculation mechanism for narrow and spread. Ink Drop acts like adding a Gaussian-like distributed noise into the learning data points. Also, narrow path and spread calculated by averaging, resulting in more immunity against noise. This section investigates this ability.

RBFN is known as one of the most powerful noise immune function approximation tools. This kernel-base network is implemented by linear combination of non-linear local decision boundaries called receptive fields. Basically, motivation for radial basis functions arises from a consideration of the interpolation problem, especially when the input variables are noisy [13]. So to probe modeling ability and noise immunity of ALMC, it is compared with RBFN to approximate 2-input, 1-output Sugeno-Yasukawa function in presence of noise. This function is described as bellow [21]:

$$y = (1 + x_1^{-2} + x_2^{-1.5})^2, \quad 1 \leq x_1, x_2 \leq 5 \quad (11)$$

In order to assess the model accuracy, the error between target function and constructed model was measured using Fraction of Variance Unexplained (FVU) defined in (12) [22]:

$$FVU = \frac{\sum_{i=1}^L (\hat{y}(x_i) - y(x_i))^2}{\sum_{i=1}^L (y(x_i) - \bar{y})^2}, \quad \bar{y} = \left(\frac{1}{L}\right) \sum_{k=1}^L y(x_k) \quad (12)$$

Where x_k is k th ($k = 1, 2, 3 \dots L$) input vector, $x_k = (x_1^k, x_2^k)$, and \hat{y} is the output of constructed model. The FVU is proportional to the mean squared error. As the model accuracy increases, the FVU approaches zero. The FVU is calculated over 10,000 mesh-like intersection points covering x_1 - x_2 domain plane. Let T_1 and T_2 be the sets of x_1 and x_2 coordinates. The input vectors of the test set are described by the Cartesian product of T_1 and T_2 , obtained from (13):

$$T_i = \left\{ t_i \mid t_i = \frac{2p-1}{200} c_1 + c_2, p = 1, 2, \dots, 100 \right\} \quad (13)$$

Table.2 Mean FVU Error of function approximation for ALMC with different training set size and partitions in comparison with RBFN.

Number of Partitions	IDS Ink Radius	Training Set Size			
		100	250	400	550
5×5	6 (%5) ⁺	0.295	0.082	0.059	0.048
	12 (%10)	0.156	0.065	0.056	0.054
	18 (%15)	0.143	0.071	0.065	0.064
8×8	6 (%5)	0.508	0.075	0.034	0.027
	12 (%10)	0.200	0.049	0.030	0.028
	18 (%15)	0.152	0.049	0.033	0.033
12×12	6 (%5)	0.991	0.136	0.041	0.024
	12 (%10)	0.440	0.056	0.028	0.025
	18 (%15)	0.289	0.051	0.031	0.029
RBFN Receptive Fields Radius					
0.01		0.365	0.063	0.033	0.025
0.0234 (%5)		0.133	0.050	0.030	0.024
0.0469 (%10)		0.128	0.060	0.046	0.045
0.0703 (%15)		0.146	0.091	0.079	0.081
0.1		0.187	0.140	0.131	0.135

⁺Percentage of IDS grid plane resolution. 256×256.

In this work $c_1 = 9$ and $c_2 = 1$ for the domain of (11). Mean FVU error over mentioned test set for ALMC, trained by four different types of training batches, has been calculated. Each data batch includes 10 random training sets with set size of 100, 250, 400 and 550 samples. The results for three ALMCs with different partitioning and three different Ink Drop patterns have been compared with RBFN in Table II.

The power of knowledge resolution tuning in ALMC is illustrated in Table II. For sparse knowledge space, small partition number and large Ink Drop lead to smaller FVU error. Adequately large Ink Drop performs better curve fitting in thinly scattered IDS plane. As it can be seen in Table II, in training data sets with 100 samples, for constant ink radius, FVU error becomes large as for increasing partition numbers. Similarly for constant partition number, increasing the ink radius reduces the FVU error. But in medium size data sets (250, 400), medium resolution (8×8 partitions, 12 Ink Drop Radius) needs to be tuned to obtain optimum FVU. Whenever knowledge space be enriched enough (data set with 550 samples), large partition number and small Ink Drop result in better performance. In this case, by dropping large IDS, spread values become large without any significant improvement in narrow trajectory extract-

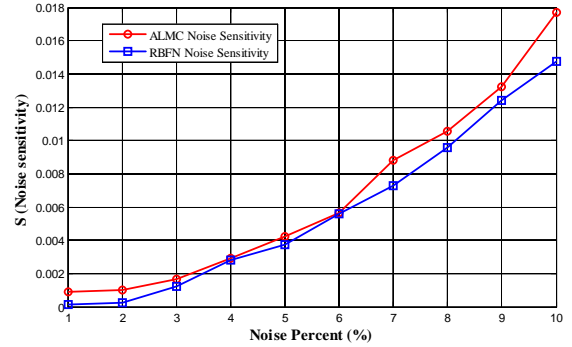


Fig.10 Noise Sensitivity of ALMC in comparison with RBFN (Spread=0.035) as Noise Percentage. “IDS radius = 9, 12×12 partitions, 256×256 IDS planes, 550 data samples”

ing. So accuracy falls down due to model over-fitting.

Regarding to ALMC Ink size, size of RBFN base function is adjusted in three tips and two additional boundary values. FVU for RBFN is comparable with optimum adjusted ALMC over most cases of data sets.

To analyze modeling performance of ALMC in noisy environment, noise sensitivity parameter is defined as following equations:

$$S = FVU_{noisy} - FVU_{noiseless} \quad (14)$$

$$FVU_{noisy} = FVU(x + \tilde{n}_p), \quad \tilde{n}_p = \frac{x \cdot \tilde{n} \cdot p}{100} \quad (15)$$

Where x is input validation vector, \tilde{n} stands for normalized uniform noise and p is noise percentage. Fig.10 describes RBFN and ALMC noise sensitivity over ten different noise percentage. It is clear that ALMC noise sensitivity is as well as RBFN which is known as one of the most famous noise robust tools for function approximation problems [13].

4. Active Classification of Remote Sensing Image

To evaluate the applicability of ALM in more realistic classification tasks, a supervised remote sensing image classification problem is chosen. This type of classification contains a large number of data points with un-balanced distribution and probably noise contents. Also, large dataset enables performance evaluation of a classifier in real applications.

Nowadays remote sensing is used in wide range applications such as agriculture and mineralogy, surveillance applications, military logistics and etc. Many researches with different directions have been

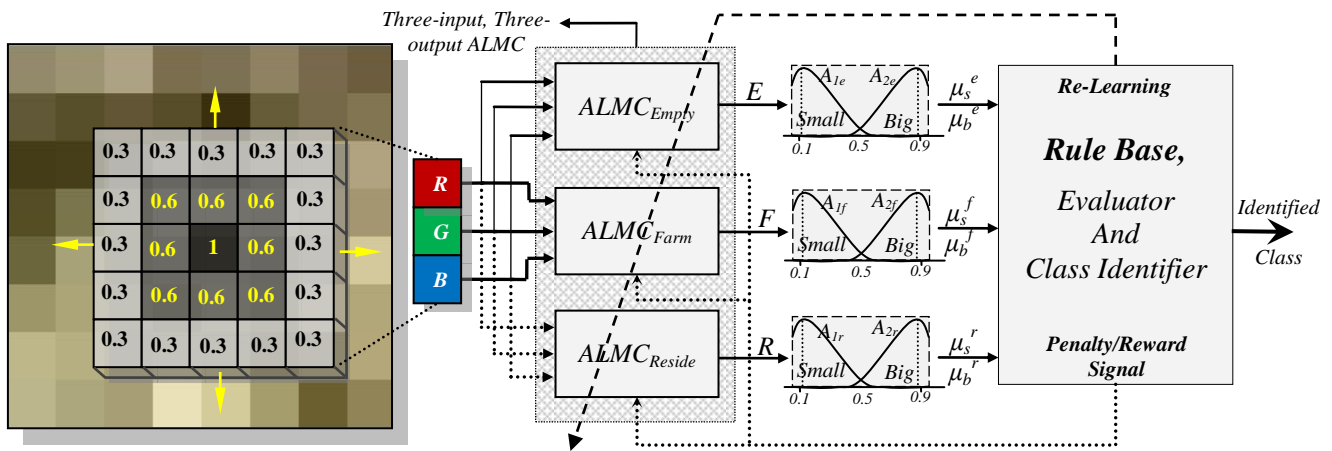


Fig.11 Left: Pyramid IDS-like operation on neighbor pixel bands acts such as bilateral filter.
 Right: General Scheme of active pattern classification by ALMC for Remote Sensing Image Classification

accomplished in this field to overcome main challenges. By dramatic growth of information processing techniques whether in soft computing or advances in hardware architectures, today it is possible to analyze Very High Resolution images with Hyper-Spectral distribution, where each pixel consists of hundreds spectral bands ranging from infrared to visible spectrum [23]. However, performance of supervised algorithms is highly pattern-dependent, so selection of proper training pattern set is emphasized [24]. But it is very time consuming and redundant to label best pixels by visual inspection and manual photo-interpretation, especially in VHR images with high dimensional Hyper-Spectral distribution. Thus, sophisticated methods for data gathering and pattern refinements are necessary and some methods has been proposed [25], [26]. These methodologies enable more accurate and less time consuming classification by preparing less redundant training sets with considerably smaller size.

Active learning as a knowledge refinement policy is a very successful approach for smart data gathering [24]. As discussed in section I, one of the leading ideas in ALMC is active learning, which has been addressed as one of the most important features of human intelligence. So it seems this classifier is inherently compatible with solution methods for the mentioned challenges in remote sensing image classification. In this section, a new ALMC architecture used as an active classifier is proposed. This classifier evaluates input patterns using fuzzy rule base evaluator, in order that unclassified or uncertain patterns could be refined by strengthening good patterns and eliminating destructive patterns through evaluator feedback signals. From another point of view, ALMC

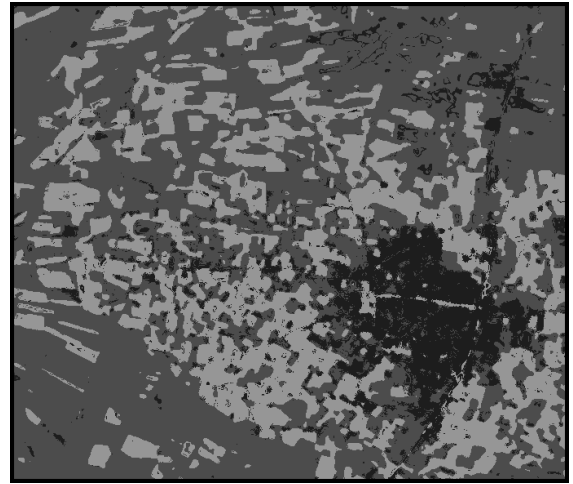
acts as an expert system evolving via a general policy. This policy has been exerted through fuzzy rule base evaluator.

In this section, ALMC is set up as three-input, three-output system (Fig.11). It comprises three independent three-input, single-output ALMCs. Each single-output ALMC tries to identify single class of terrains and distinct it from other classes. The observed pixel is labeled as 0.9 for related identifier ALMC, and 0.1 for other distinctive ALMCs. For instance, in Fig.11-right for a farm land, the output of $ALMC_{Farm}$ is marked as 0.9; $ALMC_{Empty}$ and $ALMC_{Reside}$ are labeled as 0.1. Also unclassified and unknown areas, as default output value of ALMC (section II), are naturally marked as 0.5. This style allows the evaluator to assess input patterns by analyzing belongingness and exclusion degree (μ_b^x , μ_s^x membership values in Fig.11-right) for each single class simultaneously.

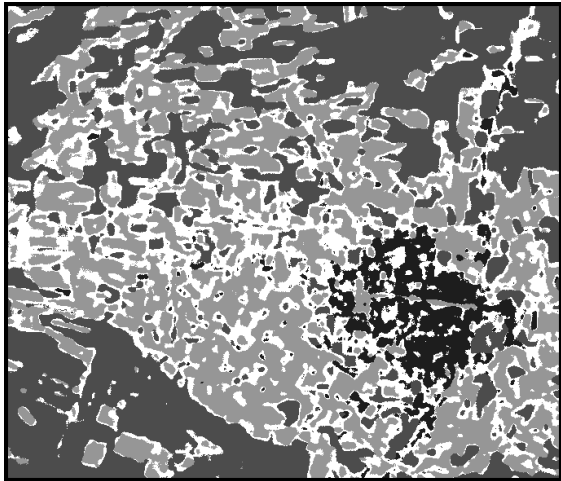
Spatial content of data has long been examined as an important feature in Geographical Information processing. Fundamentally, the first law of geography is “Everything is related to everything else, but near things are more related than distant things” [27]. So, always the neighborhood information is considered, as well as global correlation among pixels [23]. In this work, spatial content of local terrain has been considered using fuzzy IDS-like observation of neighborhood pixels. In Fig.11-left, a pyramid-form IDS window is shown, which is centered on a single pixel and performs weighted averaging on covering pixels RGB bands. This form of observation on pixel bands enables labeling in accordance with some partial information about neighborhood and texture. In addition, it acts as a bilateral filter providing better



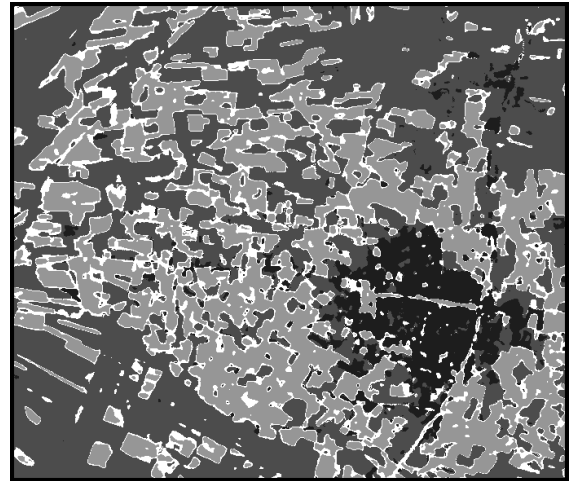
(a) Original Image



(b) ALMC



(c) SVM



(d) RBFN

Fig.12 (a): Original Satellite Image, (b): Image classified by ALMC, (c): Classified image by SVM, (d): Classified image by RBFN; Residential area is labeled as Black, Farm lands as Pale Gray; Empty lands as Dark Gray and unclassified areas have been colored by White.

noise immunity. As a result of this feature, in addition to the overlapping property and the fuzzy curve fitting technique of IDS, an input can activate single ALMCs to different degrees. It means that there is some overlapping area among class boundaries, including pixels with similar characteristic. So a fuzzy-likelihood evaluation mechanism is at hand.

The problem is identification of three types of terrains: farm lands, residential areas and empty lands in a medium resolution satellite image ($6m/pixel$, 681×584) taken by *OrbView-3*, *Geo-Eye* satellite in August 2006 from an agricultural special region of *BASTAM* fertile plain at north-east of Iran (Fig.12-a). Training data set is different from validation image, including 4500 random pixels of three different terrains. Partition numbers for ALMCs is 10 for each input domain. ALMCs are comprised of 900 IDS

units in total, with IDS plane resolution of 256 and Ink Radius of 9. Also, the radius of pyramid-IDS is set as 3 pixels, covering a local neighborhood area of $30m \times 30m$.

The main policy of the evaluator to identify input pattern is founded on belongingness and exclusion degree for each ALMCs. In other words, the best candidate for a specific class has the largest belongingness and the lowest exclusion degree for the related ALMC. Also, the patterns which are failed to classify are evaluated by a rule-base and relearned with best candidate labels. Hence, the best candidate label should be strengthened in corresponding nominated ALMC by imposing positive Ink Drop (*Reward*) coinciding negative Ink Drops in other ALMCs (*Penalty*). Evaluation process has been described as following rules (*E*, *F*, and *R* are abbrevia-

tion of Empty, Farm and Residential Lands respectively; A_{kx} denotes k -th fuzzy set of class x):

$$R_1.\text{if } E \in A_{2e} \wedge F \in A_{1f} \wedge R \in A_{1r} \rightarrow \text{Out} = E$$

$$R_2.\text{if } E \in A_{1e} \wedge F \in A_{2f} \wedge R \in A_{1r} \rightarrow \text{Out} = F$$

$$R_3.\text{if } E \in A_{1e} \wedge F \in A_{1f} \wedge R \in A_{2r} \rightarrow \text{Out} = R$$

In case of E, F, R all belonging to A_{1x} (R_4) or A_{2x} (R_5), it is necessary to find the best candidate (*lowest μ_s^x and largest μ_b^x*) and relearn pattern with new labels. Similar to R_4, R_5 , when two of E, F , and R belong to A_{2x} (R_6 - R_8), the best candidate is selected from those classes for re-labeling. The cycle of pattern analysis and imposing new patterns would be kept on until certain labeling for uncertain or unclassified areas (*hitting one of the R_1 - R_3 rules*).

In Fig.12 original image and classified images proposed by ALMC, SVM and RBFN are shown. SVM has linear kernel and LSE method is used to find the separating hyperplane. RBFN spread value is set to 0.035. Also, empty lands marked by dark-gray, farm lands are colored with pale-gray, residential area is labeled by black and white color shows unclassified areas. As shown in Fig.12, in boundary areas it is hard for SVM and RBFN classifiers to distinct each class. SVM is unable to classify 24% of area (Fig.12-c). This rate for RBFN is 13.5% (Fig.12-d), whereas the proposed active analyzing method has a clear remarkable performance in comparison with RBFN and SVM. It is noticeable that in spite of additional learning process in the proposed active classifier, overall run times are more than 3 times less than SVM and RBFN classifiers.

As discussed in this section, one of the big challenges in supervised remote sensing image classification is dealing with a large amount of ill-balanced data. But generation of a suitable training set is difficult and needs extensive manual analysis of the image. This challenge would be aggravated when image spectral dimension becomes large. Therefore, further investigations about sophisticated classifiers such as the method proposed in this paper is necessary for image analysis, especially in VHR images.

5. Conclusions

ALM is a powerful soft computing tool imitated by brain macro level findings. In this paper, ALM is investigated as an active classifier and evaluated by standard benchmarks including: Iris data set, two-spiral problem, Sugeno-Yasukawa function approxi-

mation and a remote sensing image classification. Due to the fuzzy non-exact view of ALMC on data patterns and straightforward non-iterative learning process, it is addressed in this work as a rapid noise robust classifier which could be applied in a wide range of applications. One of the interesting features of ALMC is knowledge resolution tuning that enables a convenient top-down structural evolution for classification of complex helical structures, where most of the traditional classifiers need to be modified and iterated by numerous iterations to have better performance. Also a new ALMC architecture to actively analyze a remote sensing image is proposed. The results show the viability of the proposed method in active refinement and analysis of problems with ill-balanced pattern distribution. In contrast to ANNs and SVM in which whole adjustable weights take part in recognizing and learning of single pattern, and also in contrast to ANFC, in which parameters need to be iterated by gradient descent, in ALMC only pattern-related partial knowledge participates in the learning and modeling phases. The proposed classifier is considerably fast and provides straightforward learning process with more stability and tractability.

Despite mentioned distinguished specifications of ALMC, the application of this method is highly restricted to classification problems with high dimensional input patterns such as Hyper Spectral images, due to the storage limitations for matrix-like memory of IDS units [5], [11]. We proposed an optimization solution in this work to prevent redundancy in ALMC. But IDS storage challenge seems elusive. It appears that Spike-IDS [8] as a neural implementation of IDS could overcome this challenge, using a dramatically memory efficient solution either for hardware or software platforms. Therefore, the application of Spike-IDS in high dimensional classification problems can be the subject of future research

Acknowledgment

The First Author would like to thank *Mahsa Taheran* for her revision and useful comments.

References

- [1] E. Kolman, M. Margaliot, "Knowledge-based neurocomputing: A fuzzy logic approach", Studies in Fuzziness and Soft Computing, Vol 234, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 1-5.
- [2] S. B. Shouraki, "A novel fuzzy approach to modeling and control and its hardware implementation based on brain functionality and specification", Ph.D dissertation, The

- University of Electro-Communication, Chofu-Tokyo, Japan, March 2000.
- [3] I. Singh, "Textbook of human neuroanatomy", 7th Edition, Jaypee Brothers Publishers, New Delhi, 2006, pp. 59-63.
- [4] R. DeVere, M. Calvert, "Navigating smell and taste disorders", 1th edition, Demos Medical Publishing, New York, August 2010, pp. 33-37.
- [5] M. Murakami, "Practicality of modeling systems using the IDS method: Performance investigation and hardware implementation," PhD thesis in Electrical Engineering, Department of information Technology, the University of Electro-Communication, Tokyo, Japan, March 2008.
- [6] J. Polkinghorne, "Belief in god in an age of science", Yale University Press, New Haven, USA, 1998, pp. 25-48.
- [7] S. B. Shouraki, N. Honda, "Outline of a soft computer for brain simulation (Awarded as Best Paper)", International Conference on Soft Computing Information/Intelligence Systems (IIZUKA'98), Iizuka, Japan, October 1998, pp. 545-550.
- [8] M. Firouzi, S. B. Shouraki, M. Gh. Rostami "Spiking Neural Network Ink Drop Spread, Spike-IDS", (Accepted for publication) Advances in Cognitive Neurodynamics III, Springer Science+Business Media, Netherland, November 2012, (to be published).
- [9] B. Kosko, "Fuzzy systems as universal approximators", IEEE Transaction on Computers, Vol. 43, No. 11, November 1994, pp. 1329-1333.
- [10] M. Firouzi, S. B. Shouraki, M. Tabandeh, S. H. R Mousavi, "A novel pipeline architecture of Replacing Ink Drop Spread", Proc. Second Word Congress on Nature and Biologically Inspired Computing, Kitakyushu, Japan, December 2010, pp 127-133.
- [11] F. Merrikh-Bayat, S. B. Shouraki, A. Rohani, "Memristor Crossbar-Based Hardware Implementation of the IDS Method", IEEE Transactions on Fuzzy Systems, Vol. 19, No. 6, December 2011, pp. 1083-1096.
- [12] A. J. Maren, C. T. Harston, R. M. Pap, "Handbook of neural computing applications", 1th Edition, Academic Press, San Diego, USA, December 1990, pp 328.
- [13] C. M. Bishop, "Pattern Recognition and Machine Learning", Springer, New York, USA, 2006, pp. 299-300.
- [14] M. Firouzi, "Spike-IDS, A novel biologically inspired Spiking Neural model for Active Learning Method fuzzy modeling (Modeling of human knowledge acquisition through a hybrid approach)", M.Sc Thesis in Electrical Engineering, Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran, January 2011, pp 46-48.
- [15] S. Singh, "Classification of helical structures", Proc. IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN 2000), Como, Italy, 2000, pp. 85-89.
- [16] Chihiro. Ikuta, Yoko. Uwate, Yoshifumi. Nishio, "Multi-Layer Perceptron with Glial Network for solving Two-Spiral problem", IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E94-A, No.9, 2011, pp.1864-1867.
- [17] H. Liang, G. Dai, "Improvement of cascade correlation learning algorithm with an evolutionary initialization", Information Science. Vol.112, 1998, pp. 1-6.
- [18] J.R.A lvarez-Sa´nchez, "Injecting knowledge into the solution of the two-spiral problem", Neural Computing and Applications Vol.8 Issue.3, 1999, pp. 265-272.
- [19] F. Liang, "Annealing stochastic approximation Monte Carlo algorithm for neural network training". Machine Learning, Vol. 68, August 2007, pp. 201-233.
- [20] J. S. Roger Jang, "Neuro-Fuzzy Modeling: Architectures, Analyses, and Applications", Ph.D. Dissertation, Department of Electrical Engineering and Computer Science, University of California at Berkeley, Berkeley, USA, Jul 1992, pp. 114-125.
- [21] M. Sugeno, T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling", IEEE Transactions on Fuzzy Systems, Vol. 1, No. 1, February 1993, pp. 7-31.
- [22] J. N. Hwang, S. R. Lay, M. Maechler, R. D. Martin, J. Schimert, "Regression modeling in backpropagation and projection pursuit learning," IEEE Transactions on Neural Networks, Vol. 5, No. 3, May 1994, pp. 342-353.
- [23] G. Jun, J. Ghosh, "Spatially adaptive classification of land cover with Remote Sensing data", IEEE Transactions on Geoscience and Remote Sensing, Vol. 49, No. 7, July 2011, pp. 2662-2673.
- [24] D. Tuia, M. Volpi, L. Copa, M. Kanevski, J. Muñoz-Marí, "A survey of active learning algorithms for Supervised Remote Sensing Image Classification", IEEE Journal of Selected Topics in Signal Processing, Vol. 5, No. 3, June 2011, pp. 606-617.
- [25] G. M. Foody, A. Mathur, "The use of small training sets containing mixed pixels for accurate hard image classification: Training on mixed spectral responses for classification by a SVM," Remote Sensing of Environment, Elsevier, Vol. 103, No. 2, July 2006, pp. 179-189.
- [26] D. Tuia, F. Ratle, F. Pacifici, M. Kanevski, W. J. Emery, "Active learning methods for remote sensing image classification" IEEE Transactions on Geoscience and Remote Sensing, Vol. 47, No. 7, July 2009, pp. 2218-2232.
- W. R. Tobler, "A computer movie simulating urban growth in the detroit region", Economic Geography, Vol. 46, 1970, pp. 234-240.