Unsupervised Hybrid Learning Model (UHLM) as Combination of Supervised and Unsupervised Models

Andrey Gavrilov, Sungyoung Lee
Computer Engineering Department, Kyung Hee University,
1 Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do, 446-701,
Republic of Korea

***Abstract* – In this paper novel paradigm of Unsupervised Hybrid Learning Model is proposed based on usage of unsupervised learning model as teacher for supervised learning model. This approach is result of generalization of hybrid neural model MLP-ART2, proposed by authors in [7, 8, 9]. Also we propose novel architecture of Reinforcement Learning based on our paradigm and Multilayer Perceptron (MLP). In this architecture MLP is working in two modes: attraction of output vector to target and repulsion from target with respect to award. We propose also model MLP-ART-RL based on combination of model MLP-ART2 and Reinforcement Learning.**

I. Introduction

For creating of general intelligence, in particular, for intelligent robots, learning models must satisfy to following requirements:

- Fast learning,
- Fast recalling,
- Incremental learning,
- Unsupervised learning or reinforcement learning.

Practically all models of neural networks cover just part of these features. So in last years new approach became very popular based on development of hybrid neural networks consisting of typically two different neural paradigms to get cumulative result [1, 2, 3, 4]. Such hybrid neural networks may be viewed as part of wider concept of hybrid intelligent systems consisting of different paradigms of representation and processing of knowledge [5, 6].

But really these combinations of neural models address the solving of partial technical problems for concrete application.

In this paper we propose novel approach to combine two neural models one of them is supervised learning model and another is unsupervised learning model. And these models may be variable in wide area.

This model inspired by investigations of brain and mind is a result of generalization of hybrid neural network MLP-ART2 earlier proposed by us [7, 8, 9].

II. Main Concepts Of Hybrid Unsupervised Learning Model

Our suggested hybrid model consists of two models as shown in figure 1. Model 1 is based on multi-layer neural network using error back propagation (EBP) algorithm [10]. It provides mapping of input feature space on output feature space more suitable for classification or clustering or invariant recognition. It is well known that MLP can provide arbitrary transformation of primary features [11]. So we can train one to get invariant features as outputs. The model 2 provides clustering and classification and also provides mapping of recognized class (or cluster) on output feature space of model 1 as additional result of this process. It means that it produces desirable output pattern of model 1 for algorithm EBP. The model 2 may be viewed as teacher for model 1 to adapt it to relative small transformations of input patterns. Note that unlike classical MLP the algorithm EBP in this model aims to some transforms feature space by just small number of iterations but no reduction of error to very small value.

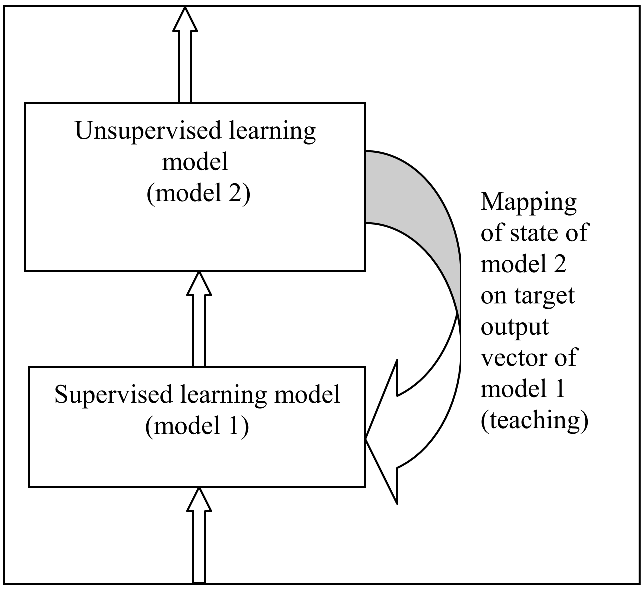

Fig. 1. Structure of proposed Unsupervised Hybrid Learning Model

III. Some Implementations Of Unsupervised Hybrid Learning Model

A. Model MLP-ART2

One implementation of this paradigm was proposed and investigated in [7, 8, 9]. This model consists of multi-layer perceptron with error back propagation (EBP) as model 1 and ART-2 [12] as model 2.

MLP provides conversion of primary feature space to secondary feature space with lower dimension and with more suitable features (close to invariant features) for clustering. Neural network ART-2 classifies images and uses secondary features to do it. Training of MLP by EBP (with limited small number of iterations) provides any movement of an output vector of MLP to centre of recognized cluster of ART-2 in feature space. In this case the weight vector (center) of recognized cluster is desired output vector of MLP. It could be said that the recognized class is a context in which system try to recognize other images as previous, and in some limits the system "is ready to recognize" its by this manner. By other words neural network "try to keep recognized pattern inside corresponding cluster which is recognizing now".

Action of the suggested model is described by the following unsupervised learning algorithm:

1. In MLP let the weights of connections equal to *1/n,* where *n* is quantity of neurons in the previous layer (number of features for first hidden layer). The quantity of output neurons N_{out} of ART-2 is considered equal zero.

2. The next example from training set is presented to inputs of MLP. Outputs of MLP are calculating.

3. If N_{out}=0, then the output neuron is formed with the weights of connections equal to values of inputs of model ART-2 (the outputs of MLP).

4. If N_{out}> 0, in ART-2 the algorithm of calculation of distances between its input vector and centers of existing clusters (the weight vectors of output neurons) is executing using Euclidian distance:

$$d_j = \sqrt{\sum_i (y_i - w_{ij})^2}$$

Where: y_i – i^{th} feature of input vector of ART-2, w_{ij} – i^{th} feature of weight vector of j^{th} output neuron (the center of cluster). After that the algorithm selects the output neuron-winner with minimal distance. If the distance for the neuron-winner is more than defined a vigilance threshold or radius of cluster *R*, the new cluster is created as in step 3.

5. If the distance for the neuron-winner is less than *R,* then in model ART-2 weights of connections for the neuron-winner are recalculated by:

$$w_{im} = w_{im} + (y_i - w_{im})/(1 + N_m)$$

Where: N_m – a number of recognized input vectors of m^{th} cluster before. In addition, for MLP a recalculation of weights by algorithm EBP is executing. In this case a new weight vector of output neuron-winner in model ART-2 is employed as desirable output vector for EBP, and the quantity of iterations may be small enough (e.g., there may be only one iteration).

6. The algorithm repeats from step 2 while there are learning examples in training set.

Note that in this algorithm EBP aims at absolutely another goal different from that in usual MLP-based systems. In those systems EBP reduces error-function to very small value. But in our algorithm EBP is needed only for some decreasing distance between actual and desirable output vectors of MLP. So in our case the long time learning of MLP is not required.

Note that EBP and forming of secondary features are executed only when image "is captured" by known cluster. So selection of value for vigilance threshold is very important. We used different heuristics for calculation of this parameter and adaptation of it to characteristics of images.

This model was tested in different experiments with processing of sequence of visual images [7, 8, 9] and with avoidance of obstacles by simulated mobile robot [13, 14]. Experiments show that this model provides the invariant recognition of images and may be applied for image processing and pattern recognition in dynamic environment.

B. Model MLP-RL

Another variant of model based on proposed paradigm may be combination of MLP and Reinforcement Learning (RL) [15]. There are many of different reinforcement learning algorithms. Well known are especially Q-learning algorithm and actor-critic architectures [16]. We use in this paper last kind of RL. We focus to application of RL in mobile robots.

In this case the model 2 is critic producing positive or negative award (the punishment) with respect to output of model 1. Model 1 may be viewed as the model for producing the actions like response to any situation described by input vector got from sensors. Positive award causes the interpretation of output of model 1 as attractive output vector (desirable action in any situation) and the punishment is corresponding with repulsive output vector. After action of model 2 the output vector of MLP becomes target vector (positive or negative). May be possible to call negative target vector as danger vector.

For this case it is needed MLP with 2 modes of EBP – positive or negative respecting for attraction and repulsion of target output vector. Let call this kind of EBP as Error Back Propagation with Punishment (EBPP). Positive mode of this model is classic EBP. The negative mode provides update of weights with opposite sign. Thus updates of weights in EBPP are described by following formulas:

$$\Delta w_{ij} = ar\varphi_j x'_i,$$

where:
w_{ij} is weight of connection between *i*th neuron and jth neurons;
a is value of award, 1 or -1;
r is a rate of learning;
φ_j is error propagation for *j*th neuron;
x'_i is derivative of active function of *i*th neuron.

Function φ_j for calculation of error propagation for output layer differs from same function in usual EBP algorithm. For case *a=1* it is same as in EBP classic algorithm:

$$\varphi_j = y_j(1-y_j)(d_j - y_j),$$

where y_j and d_j are actual and desirable output of neuron respectively.

For case $a=-1$ the function φ_j is determined as

$$\varphi_j = ky_j(1-y_j)\exp[-\frac{1}{2\sigma^2}(d_j - y_j)^2]$$

The expression $y_j(1-y_j)$ of this formula represents the derivative of neuron's state like in usual error back propagation. The exponential function in this formula provides maximal value of φ_j at equality of actual and desirable states of jth neuron. Value σ represents the sensitivity in neighborhood of danger vector. Coefficient k may be interpreted as a level of timidity and may be connected with simulation of emotions.

Another variant of calculation of φ_j is possible.

For $d_j \neq y_j$ function φ_j may be represented as

$$\varphi_j = \frac{ky_j(1-y_j)}{d_j - y_j}.$$

For $d_j = y_j$ function φ_j may be determined as constant value k.

Unlike classical EBP with positive award the punishment in EBPP provides adaptation of weights to repulsion of target output vector (danger vector). Particular case is learning to predict of events in time. In this case MLP may be replaced by recurrent neural network (model RNN-RL) dealing with sequences of patterns, e.g. Elman model [17] with EBP through time.

To provide the adaptation of weights of MLP with EBPP it is needed to use MLP in regime of training. This regime may run during detection of award or during any longer time. To support last opportunity we can utilize a memory for storing last output vector of MLP associated with the award and usage of training regime with this target vector during some s steps. This is similar to simplest behavior of animal providing the storing of attraction or repulsion in neighborhood of any target. More sophisticated model may be implemented with memory for all met attractive and repulsive output vectors of MLP and recognition of most close of them for example by ART-2. In this case we have combination of MLP with EBPP, ART-2, critic and memory of target vectors and corresponding award (model MLP-ART-RL). This model is shown in figure 2. Algorithm for this model in pseudo code is represented below.

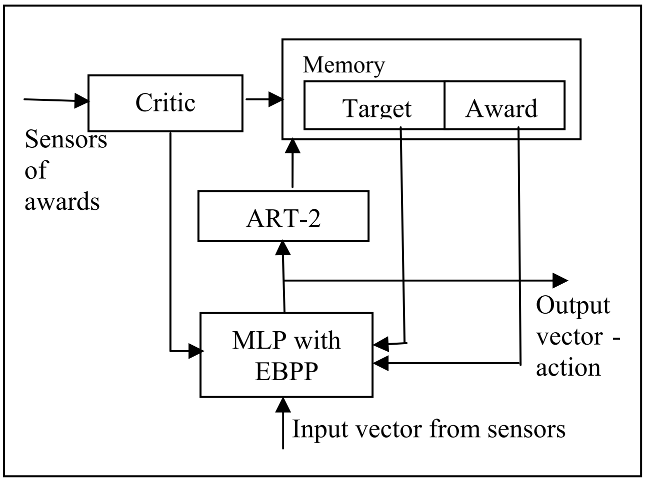

Fig. 2. Structure of model MLP-ART-RL

Procedure processing input vector.

```
Calculation of outputs of MLP.
Calculation of outputs of ART-2 (distances between input
vector of ART-2 and centers of existing clusters);
If minimal value of outputs of ART-2 > r
  Then
    If in previous step new cluster was created
          with same award
      Then
        r = minimal value of outputs of ART-2
      Else
        If award or punishment is detected
          Then
            Target output of MLP is actual output vector
                of MLP
            Create new record in memory.
            Create new cluster (output neuron) for ART-2
              with center equal target output vector of MLP.
            Store target output vector and award
                in new record.
          Else
            r = minimal value of outputs of ART-2
        End if
    End if
End if
If minimal value of outputs of ART-2 ≤ r
   Then
    Update weights of ART-2.
    Search recognized cluster (record) in memory and read
        the award.
    If award is positive
      Then
        Train MLP with EBP
      Else
        Train MLP with EBPP.
    End if
End if
```

End of procedure.

Note that unlike usual ART-2 in this algorithm vigilance threshold *r* is changed adapting to process of creating or recognition of clusters. In beginning this parameter must have small value. And in this model (based on ART-2) is not needed to store in memory the target vector. This one is storing in ART-2 as weight vector of output neuron (cluster). Note also that it is needed to introduce any delay between calculation of output of MLP and detection of award or punishment or outputs of MLP must me obtained in previous step.

IV. Experiments with model MLP-ART-RL

Previous experiments with the model MLP-ART2 [8, 9] show that most sufficient problem is selection value of vigilance threshold *r*. Proposed and described above algorithm provides adaptation of this value to features of recognition and creating of clusters in time. In case when any cluster is not recognized and in previous step new cluster was created with same award as current, value of r became equal minimal found distance between it and center of any existing cluster. This provides recognition of this cluster. In case when previous condition is fail and award or punishment is not detected value of r is setting by same way. This heuristic protects from reiteration of making similar clusters with same award and repetition of action without any award.

This algorithm was tested in simulated mobile robot in program MRS already used earlier by authors [13, 14]. In this experiments, as in previous ones, the hybrid approach was used, i.e. this proposed model was used at enough distance between robot and any obstacle and the simple rules were used near the obstacles. And also when robot is enough far from target and obstacles then it selects the direction of movement to target. In these experiments the robot got positive award when next position of robot was closer to target or the robot could look directly the target not covered by any obstacle.

The robot got the punishment when next position was more far from target in contrary to previous one. Actions of robot were provided by 4 outputs of MLP: 1) keep the direction of movement, 2) turn to left in 30 degrees, 3) turn to right in 30 degrees, 4) turn to 180 degrees. The input vector of MLP have length 15 and consists of information from 12 distance sensors around body, the distance from target, the direction to target and the direction of robot.

Typically behavior of robot in experiment is shown in figures 3 and 4. There yellow positions (more light) mean situations in which robot creates new cluster and stores corresponding award. In figure 3 there are two such positions with award and punishment respectively (one of them was covered later by another position and so is invisible). In figure 4 there are two such positions with only awards. At repetition of movement of learned robot from new start position usually the clusters created in previous experiment are enough and new ones are not needed for achievement of target.

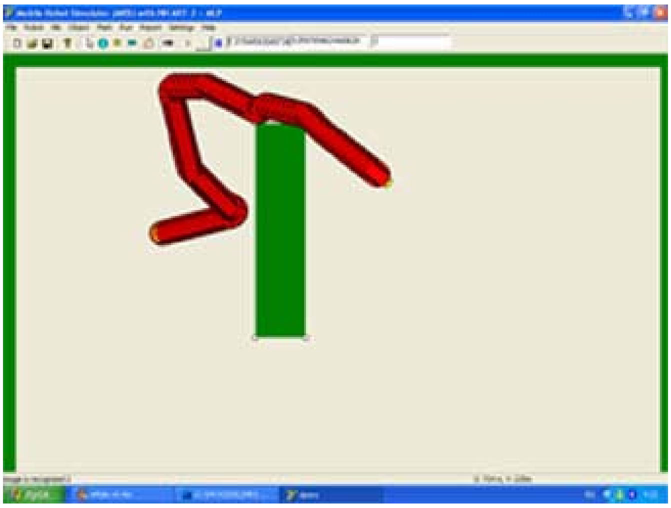

Fig. 3. The behavior of mobile robot in environment with one obstacle controlled by hybrid navigation system based on proposed model.

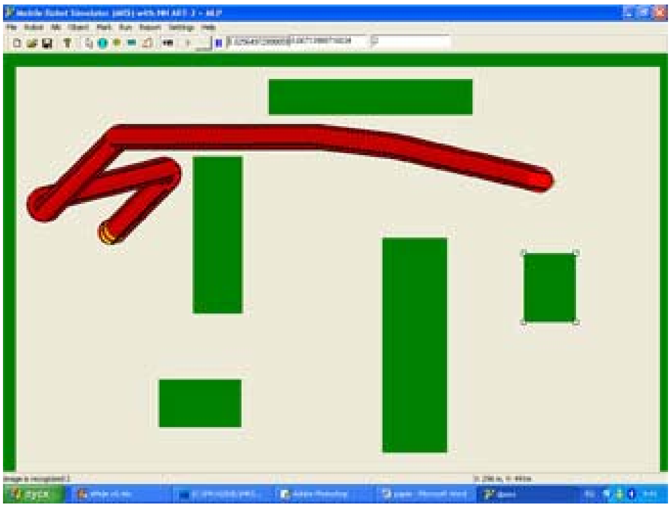

Fig. 4. The behavior of mobile robot in environment with multiple obstacles controlled by hybrid navigation system based on proposed model.

The experiments using proposed model MLP-ART-RL were compared with experiments utilizing multi layer perceptron without ART, memory and punishment. In contrast to MLP our model provides less chaotic and more "reasonable" behavior with small number of iterations of error back propagation (e.g. for 1 or 5). It is caused by invariant recognition inherited from model MLP-ART and including of punishment when robot moves off target. In case of large number of iteration (e.g. 50 or more) MLP demonstrates approximately same capabilities but has any disadvantages: 1) more time for training, 2) possibility of overfitting and so impossible relearning and 3) difficulty of connection with human-robot interaction by natural language. The last disadvantage is absent when we use clustering by ART as was shown in [18] where was proposed the architecture of control system for mobile robot oriented on natural language based interaction with him for definition of targets of movement.

In comparison of previous our experiments with robot controlled by MLP-ART2 new model MLP-ART-RL provides fully unsupervised learning whereas MLP-ART2 required sometimes assistance of operator for creating of new cluster with respective direction of desirable movement.

Conducted experiments show that behavior of the robot controlled by proposed model is better than movement controlled by multi layer perceptron. Advantage of this model in comparison with classical reinforcement learning is absence of necessity of preliminary knowledge about environment and discretization of space.

V. Related works

Our work is related with research in invariant recognition and usage of neural networks for navigation of mobile robots.

There are many other approaches to achieve invariant recognition by neural networks, for example proposed in [19], [20], [21], [22], [23], [3], [4]. In [19] K. Fukushima suggests complex universal neural model for invariant recognition "Neocognitron". But the complexity of this model does not permit to use one in real time. In [20] the combination of wavelet transform and the Fourier transform was proposed for building of so called spectroface for invariant recognition in particular face recognition. In [21] was proposed invariant recognition based on preprocessing (extraction of topological features and calculation of moment of inertia) and holographic nearest-neighbor algorithm and this approach was tested by character recognition.

Thus each of proposed earlier models for invariant recognition is either too complex or specialized for determined any kind of images and transformations.

We suggest potentially universal approach for invariant recognition which can be implemented in real time systems, because it not requires long time processing as in usual applications of error back propagation.

Usage of neural networks for navigation of mobile robots is very popular in robotics from works of N.M. Amosov [24] and R. Brooks [25]. Short review of this topic may be found in [26]. This interest of using neural networks for this problem is explained by that a key challenge in robotics is to provide the robots to function autonomously in unstructured, dynamic, partially observable, and uncertain environments. The problem of navigation may be divided on following tasks: map building, localization, path planning, and obstacle avoidance. Our work is dealing with obstacle avoidance and path planning.

Many attempts to employ different neural networks models for obstacle avoidance are known, for example [27-30].
Usage of multi layer perceptrons (MLP) with error back propagation learning algorithm has some disadvantages most of them are complexity or even impossibility to relearn, slow training and orientation on supervised learning. In [27] was made the attempt to overcome some of these shortcomings by development of multi layer hybrid neural network with preprocessing based on principle component analysis (PCA). This solution allows some reduce the time of learning. But rest disadvantages of MLP are remained. In [28] A. Billard and G. Hayes suggested architecture DRAMA based on recurrent neural network with delays. This system is interesting as probably first attempt to develop universal neural network based control system for behavior in uncertain dynamic environment. However it was oriented on enough simple binary sensors detecting any events. The attempt of employ of model ART-2 for solving of navigation task of robot oriented on interaction by natural language was carried out in [18]. But this model is dealing with primary features of images and so is sensitive to its transformations. So it is difficult to use ART-2 in dynamic unknown environment. Same difficulty is expected for other popular modifications of ART (but with supervised learning), such as ARTMAP [29], Fuzzy ARTMAP [30]. To overcome this drawback in [31] was employed multi-channel model and evaluated for solving of minefield navigation task. But in this model for every category is needed to use separate ART model. This feature limits availability of such approach, essentially in case of using visual-like sensor information.

Analyzing existing approaches we may conclude that our model MLP-ART-RL is most appropriate for building of hybrid control system for mobile robot because one provides invariant recognition of environment, unsupervised learning and opportunity of categorization which may be used for association with words of natural language for dialog.

VI. Conclusions

The simulation of above described models based on Unsupervised Hybrid Learning Model (UHLM) shows its possibility of application for invariant recognition in uncertain changing environment (MLP-ART2), for learning by awards and punishments (demanding repulsion) (MLP-RL and MLP-ART-RL). These models may be used for mobile robots acting in unknown dynamic environment. Moreover this model may be combined with supervised learning of MLP as part of hybrid model. So this approach to building of hybrid neural model may be viewed as possible model of mind.

We belief that our approach implemented in UHLM may be used for other models of neural networks, for example, for building hybrid model MLP-SOM, combining MLP and self-organizing maps of T.Kohonen [32]. It is possible to expect the more stability for mapping of sensor information to output neurons. But note that in this case map probably can not use for visualization of sensor information but may use for visualization of secondary features that more invariant.

Acknowledgment

This work was supported by MIC Korea (IITA Visiting Professorship Program). Dr. S.Y. Lee is the corresponding author.

References

[1] Shie-Jue Lee and Chun-Liang Hou, "An ART-Based Construction of RBF Networks," *IEEE Trans. Neural Networks*, 13(6), 2002, pp. 1308-1321.

[2] R.A. Baxter, “Supervised Adaptive Resonance Networks,” *in Proc. of the Conf. on Analysis of Neural Network Applications*, ACM, VA, Fairfax, 1991, pp. 123-137.

[3] Jung-Hua Wang, Jen-Da Rau and Wen-Jeng Liu, “Two-Stage Clustering Via Neural Networks,” *IEEE Trans. Neural Networks*, 14(3), 2003. pp. 606-615.

[4] Hahn-Ming Lee, Chih-Ming Chen and Yung-Feng Lu, “A Self-Organizing HCMAC Neural-Network Classifier,” *IEEE Trans. Neural Networks*, 14(1), 2003, pp. 15-27.

[5] L.R. Medsker, *Hybrid Intelligent Systems*, Kluwer, 1995.

[6] M. Funobashi, A. Moeda, Y. Morooka and K. Mori, “Fuzzy and Neural Hybrid Expert Systems: Sinergetic AI,” *AI in Japan*, IEEE, 1995, August, pp. 33-40

[7] A.V. Gavrilov, “Hybrid neural network based on models of multi-layer perceptron and adaptive resonance theory,” *in Proc. of 9th Korean-Russian Int. Symp. on Science and Technology KORUS-2005,* Novosibirsk, Russia, June, 2005, NSTU, pp. 119-122.

[8] A.V. Gavrilov, Y.-K. Lee and S.-Y. Lee, “Hybrid neural network model based on multi-layer perceptron and adaptive resonance theory,” *in Proc. of Third Int. Symp. on Neural Networks ISNN-2006*, part 1, Chengdu, China, May, 2006, Springer, Advances in Neural Networks ISNN-2006, LNCS 3971, pp. 707-713.

[9] Andrey Gavrilov and Sungyoung Lee, “An Approach for Invariant Clustering and Recognition in Dynamic Environment,” *In Proc. of IEEE Int. Conf. CISSE-2006*, December, 2006.

[10] D.E. Rumelhart, “Parallel Distributed Processing,” in *Explorations in the Microstructure of Cognition*, vol. I, II, Mcclelland J.L. (Eds.), MIT Press, MA, Cambridge, 1986.

[11] Alexander N. Gorban and Donald Wunsch, ”The General Approximation Theorem,” *in Proc. IEEE / INNS International Joint Conference on Neural Networks ’98, IEEE World Congress on Computational Intelligence*, vol. 1, Anchorage, Alaska, May, 1998, pp. 1271-1274.

[12] G., A. Carpenter and S. Grossberg, *Pattern Recognition by Self-Organizing Neural Networks*, Cambridge, MA, MIT Press, 1991.

[13] Andrey Gavrilov and Sungyoung Lee, “An Architecture of Hybrid Neural Network based Navigation System for Mobile Robot,” *in Proc. of 7th Int. Conf. on Intelligent Systems Design and Applications ISDA’07*, Brasil, October, 2007 (in press).

[14] Andrey Gavrilov and Sungyoung Lee, “Usage of Hybrid Neural Network Model MLP-ART for Navigation of Mobile Robot,” *in Proc. of Int. Conf. on Intelligent Computing ICIC’07*, China, 21-24 August, 2007, LNAI 4682, pp. 182-191.

[15] Richard S. Sutton and G. Barto, *Reinforcement Learning (An Introduction),* MIT Press, 1998.

[16] Paul J. Werbos, “Stable adaptive control using new critic designs,” *http://xxx.lanl.gov/html/adap-org/9810001*, Sep 1998.

[17] J.L. Elman, “Distributed representations, simple recurrent networks, and grammatical structure,” *Machine Learning*, 7(2/3), 1991, pp. 195-226.

[18] A.V. Gavrilov, V.V. Gubarev, K.-H. Jo and H.-H. Lee, “Hybrid Neural-based Control System for Mobile Robot,” *in Proc. of Int Symp. KORUS-2004*, Tomsk, 2004, vol. 1, pp. 31-35.

[19] K. Fukushima, “Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,” *Biol. Cybern.,* vol. 36, 1980, pp. 193-201.

[20] Jian Huang Lai, Pong C. Yuen and Guo Can Feng, “Face recognition using holistic Fourier invariant features,” *Pattern Recognition,* vol. 34, 2001, pp. 95-109.

[21] L. A. Torres-Méndez, J. C. Ruiz-Suárez, Luis E. Sucar, and G. Gómez, “Translation, rotation, and scale-invariant object recognition,” *IEEE Tran. on Systems, Man and Cybernetics - Part C: Applications and Reviews*, vol. 30(1), 2000, pp.125-130.

[22] V. Petridis and V.G. Caburlasos, “Fuzzy lattice neural network (FLNN): a hybrid model for learning,” *IEEE Trans. on neural networks*, vol. 9(5), 1998, pp. 877-890.

[23] Satoshi Suzuki and Hiroshi Ando, “A modular network scheme for unsupervised 3D object recognition,” *Neurocomputing,* vol. 31, 2000, pp. 15-28.

[24] N.M. Amosov, E.M. Kussul and V.D. Fomenko, “Transport robot with a neural network control system,” *Advance papers of the Fourth Intern. Joint Conference on Artificial intelligence,* 9, 1975, pp. 1-10.

[25] R. Brooks, “A robust system layered control system for a mobile robot,” *IEEE Trans. on robotics and automation,* RA-2, 1986, pp. 14-23.

[26] An-Min Zou, Zeng-Guang Hou, Si-Ya Fu and Min Tan, “Neural Network for Mobile Robot Navigation: A Survey,” *In Proceedings of International Symposium on Neural Networks ISNN-2006*, LNCS 3972, Springer-Verlag, Berlin, Heidelberg, New York, 2006, pp. 1218-1226.

[27] D. Janglova, “Neural Networks in Mobile Robot Motion,” *International Journal of Advanced Robotic Systems,* 1(1), 2004, pp. 15-22.

[28] A. Billard and G. Hayes, “DRAMA, a Connectionist Architecture for Control and Learning in Autonomous Robots,” *Adaptive Behavior,* 7(1), 1999, pp. 35-63.

[29] G.A. Carpenter, S. Grossberg and J.H. Reynolds, “ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network,” *Neural Networks,* 4(5), 1991, pp. 565-588

[30] G.A. Carpenter, S. Grossberg N. Marcuzon, J.H.Reynolds and D.B. Rosen, “Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps,” *IEEE Transactions on Neural Networks*, 3(5), 1992, pp. 698-713

[31] Ah-Hwee Tan, “FALCON: A fusion architecture for learning, cognition and navigation,” *In Proceedings of IEEE International Joint Conference on Neural Networks IJCNN04*, vol. 4, pp. 3297-3302.

[32] T. Kohonen, “Self-organized formation of topologically correct feature maps,” Biological Cybernetics, vol. 43(1), 1982, pp. 59-69