ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 33-38

Neural Networks Organizations to Learn Complex Robotic Functions

Gilles HERMANN, Patrice WIRA, Jean-Philippe URBAN,
MIPS-TROP Research Group, University of Mulhouse

4 rue des Frères Lumière, 68093 Mulhouse FRANCE
{g.hermann; p.wira; jp.urban}@uha.fr

Abstract. This paper considers the general problem of function estimation with a modular approach of neural computing. We propose to use functionally independent subnetworks to learn complex functions. Thus, function approximation is decomposed and amounts to estimate different elementary sub-functions rather than the whole function with a single network. This modular decomposition is a way to introduce some *a priori* knowledge in neural estimation. Functionally independent subnetworks are obtained with a bidirectional learning scheme. Implemented with self-organizing maps, the modular approach has been applied to a robot control problem, a robot positioning task.

1 Introduction

Artificial Neural Networks (ANN) have proven to be very good function estimators. Unfortunately, in the case of real-world complex functions, their implementation using a typical ANN is not trivial. This is due to the high dimensionality of the problem: learning is not easy, convergence not guaranteed, and response time may become prohibitive. We propose a modular approach to address this problem. Modular neural systems are considered in terms of collections of subnetwork modules. A review of the different concepts of modularity is proposed in [3]. Two approaches for generating such networks are distinguished.

The first approach is based on algorithms that can actually generate networks and subnetworks, topologies, connections, and weights to satisfy specific constraints. This approach encloses networks of networks. A neural system is then viewed as a hierarchy of networks. Each unit from the high-level network is associated to a subnetwork to subdivide the problem to reach an adequate fineness. This structure is generally well suited to simple decompositions and solve typically classification problems.

The second approach includes networks with functionally independent subnetworks, where each subnetwork is designed to have specific functions, com-

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 33-38

munication, and adaptation characteristics. This approach includes two types of decomposition, the *mixture of experts* approach [5] and functionally independent subnetworks.

Mixtures of experts are particularly popular, and consist in using several modules in parallel and in choosing the best fit for a given environment. A similar concept is developed in [4]. This architecture is based on multiple pairs of forward and inverse models, where each pair is learned for a given control task . Simultaneously, "how to select" the set of appropriate models for a given environnement is also learned. This type of organization does not accept *a priori* information.

On the contrary, functionally independent subnetworks are compatible with prior knowledge about the function to learn. This modular approach of neural computing amounts to decompose the function and to estimate different elementary sub-functions. As an illustration, a bidirectional learning architectures is proposed in [6]. This concept allows any complex function to be decomposed into subfunctions and to build internal representations.

We adopted this modular architecture for its proven robustness [2] and its capabilities to integrate *a priori* knowledge. The modular architecture is implemented with Self-Organizing Maps (SOM) neural networks associated to Linear Local Maps (LLM). Each module is then an independent SOM-LLM network.

In Section 2, we describe a modular bidirectional learning principle based on SOM-LLM. Section 3 gives a formulation of the robot control problem and validates the modular learning for that application. Section 4 reports some of our simulations. Finally, Section 5 summarizes the paper and gives some thoughts about future studies.

2 Modular Neural Learning

In modular learning, the problem is the estimation of a transformation $\mathbf{f}$ between two spaces, $\mathcal{X} \in \mathbb{R}^n$ and $\mathcal{Y} \in \mathbb{R}^m$:

$$\mathbf{f} : \mathcal{X} \longrightarrow \mathcal{Y},\ \mathbf{y}_k = \mathbf{f}(\mathbf{x}_k). \tag{1}$$

Each neural decomposition is associated to a given application. As an example, we will expose the general case of a sequential decomposition where each module is composed of a SOM-LLM neural network. In a sequential neural decomposition, the problem is decomposed in two serial modules A and B, as represented by Figure 1. Decomposing any function or transformation in two serial modules undeniably leads to estimate an internal space. Problems appear with supervised modules and when there are no available data to supervise the two learning processes.

To overcome the problem, a bidirectional learning scheme is implemented by introducing a new module in the architecture. In Figure 1, module C is associated to module B. This new module takes its inputs from the output space $\mathcal{Y}$. While module A gives an estimation $\mathbf{z}_k^A$ of the internal representation

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 33-38

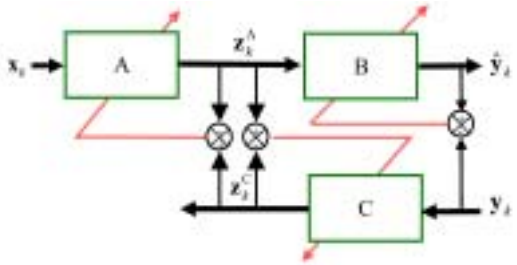

Figure 1: A sequential bidirectional learning architecture

from the input space $\mathcal{X}$, module C delivers another estimation of the internal representation, $\mathbf{z}_k^C$, from the output space $\mathcal{Y}$. Both estimations, $\mathbf{z}_k^C$ and $\mathbf{z}_k^A$, of the internal representation are used to define error signals to supervise module A and module C. The estimation $\mathbf{z}_k^A$ serves as input to module B to deliver the global estimation $\hat{\mathbf{y}}_k$. Stability and convergence of this modular architecture is analyzed in [1].

3 Application to robot control

3.1 Context

Our application consists in the visual servoing of a three-degrees-of-freedom robot-arm in its 3-D space. An active vision scheme is used as feedback: two cameras mounted on a four-degrees-of-freedom robotic head follow the robot's end-effector without necessarily being centered. The two cameras are controlled separately. In the following experiments, each camera independently rotates along both pan and tilt axes.

The two cameras define the visual feature space, $\mathcal{V} \in \mathbb{R}^4$. At each iteration k, the image processing system issues the robot's end-effector position $\mathbf{v}_k \in \mathcal{V}$ and the target position $\mathbf{d}_k \in \mathcal{V}$.

The visual features are function of the orientations of the image planes, thus of the joint angle articulations $\boldsymbol{\phi}_k \in \boldsymbol{\Phi}$ of the robotic head. $\boldsymbol{\Phi} \in \mathbb{R}^4$ is the head joint angle space.

Considering $\boldsymbol{\Theta} \in \mathbb{R}^3$ as the robot manipulator joint angle space and $\boldsymbol{\theta}_k \in \boldsymbol{\Theta}$, the joint angles vector, we can define the forward kinematic transformation $\mathbf{f}$:

$$\mathbf{f} : \boldsymbol{\Theta} \longrightarrow \mathcal{V},\ \mathbf{v}_k = \mathbf{f}(\boldsymbol{\theta}_k, \boldsymbol{\phi}_k). \tag{2}$$

The robot's end-effector $\mathbf{v}_k$ is a function of the angular joint value $\boldsymbol{\theta}_k$ and a function of the cameras' orientation $\boldsymbol{\phi}_k$.

The robot control consists in computing its joint angle corrections from image measurements and in canceling an error signal in the visual feature space. This control is fulfilled by estimating the inverse $\mathbf{f}^{-1}$ of the aforementioned

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 33-38

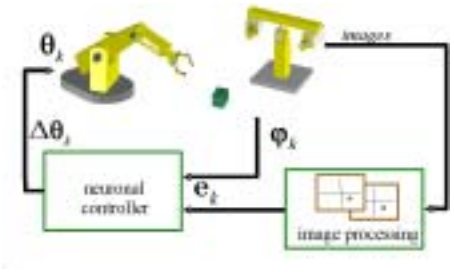

Figure 2: The principle of visual servoing

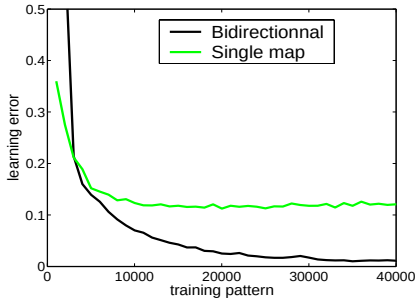

Figure 3: The neural responses for estimating the robot's joint angles

transformation with learning techniques. It is referred to as adaptive visual servoing.

3.2 Robot-Vision Modular Learning

The decomposition is now applied to estimate the inverse of the kinematic transformation defined in (2). Robot functions are typically complex, multidimensional and highly nonlinear. We propose to use prior knowledge to reduce the complexity of the transformation into sub-functions. A sequential decomposition is appropriate because it allows the estimation of an internal space with an objective representation: the space of head joint angle for centered visual features.

Let $\hat{\boldsymbol{\phi}}_k^C$ be the particular camera configuration that centers a point in 3-D space. A point is considered as centered when its image appears at the center of the image plane for both cameras. $\hat{\boldsymbol{\phi}}_k^C$ is unique for a given point in the 3-D space and can thus be used in the decomposition of $\mathbf{f}^{-1}$. Any non-centered point in the 3-D space, defined in the sensor space (the input space of the neural network) by vectors $\mathbf{v}_k$ and $\boldsymbol{\phi}_k$, respectively the visual features and the camera configuration, can be represented by a centered camera configuration $\hat{\boldsymbol{\phi}}_k^C$, a vector of smaller dimension.

The estimation of this vector is used to build an accurate robot controller. We propose to use the sequential decomposition presented in Figure 1 to build an internal space which is a representation of the image coordinates with centered cameras.

In this architecture, each module is a SOM-LLM neural network. The rule of module A is to bring any non-centered point defined by the couple of vectors $\mathbf{v}_k$ and $\boldsymbol{\phi}_k$ into the space of the centered camera configuration, e.g., to estimate the camera configuration $\hat{\boldsymbol{\phi}}_k^C$ that will center it. Module C returns another estimate of the centered joint angles and the difference between both estimates is used as a signal error to adapt the weights of module A and C.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 33-38

Module B takes the estimated signal issued from Module A to provide $\boldsymbol{\theta}$, the angular joint value of the robot-arm.

4 Results

Using an active vision system highly complicates the control problem. Indeed, the three degrees of freedom defining the robot function are complemented with the four degrees of freedom of the active vision sensor. In this set of simulations, we compare the performances of a single SOM-LLM to those of the proposed bidirectional modular learning scheme with approximately the same number of neurons.

The single SOM-LLM controller is a 3-D map of $12\times9\times12 = 1296$ neurons. The six inputs of this map are the visual features extended with the cameras' orientations, i.e. vector $[\ \mathbf{v}_k \quad \boldsymbol{\phi}_k\ ]^T$, and its desired outputs remain the robot-arm joint angles. In the bidirectional architecture, module A is composed of three 1-D maps each of size $12\times1\times1$. Modules B and C are both 3-D maps of size $9\times6\times9$. Thus, with 1008 neurons, the global modular architecture uses a number of neurons close to this of the single SOM-LLM.

The curves in Figure 3 show the better learning performances of the modular approach. The responses of the neural modular system are also evaluated by comparing the resulting 3-D end-effector positions to the desired position. The distance between these two positions is represented in Table 1 for the bidirectional modular architecture and a single SOM-LLM. The control loop using the bidirectional modular architecture results in positioning the robot's end-effector with a precision of 1.7 mm in the 3-D space. The modular architecture is thus favorably compared.

The single SOM-LLM suffers from two essential drawbacks: the difficult deployment of the 6-D input map, and the convergence of the learning algorithm. The modular approach overcomes these difficulties with its objective and coherent decomposition of the complex problem in subproblems.

The use of a modular system allows individual modules to participate in motor learning without affecting the motor behaviors already learned by other modules. Such modularity can therefore speed up motor learning while retaining previously learned behaviors.

Controller	Positioning error in the 3-D space in mean (mm)		
	X (max.)	Y (max.)	Z (max.)
SOM-LLM	6.536 (33.044)	14.178 (143.869)	13.643 (71.065)
Bidirectional	1.410 (9.992)	0.469 (4.038)	0.742 (7.785)

Table 1: The precision in positioning tasks with oriented cameras over a set of 20 simulations (80000 learning data and 10000 targets to reach)

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 33-38

5 Summary and conclusion

In this paper, we propose functionally independent subnetworks, based on a bidirectional learning scheme, to learn complex functions. The problem is decomposed and amounts to estimate elementary sub-functions with SOM-LLM modules. As an example, the inverse kinematic of a robot transformation is decomposed and learned. The resulting estimation serves to control a robot in a positioning task. An image-based visual servoing robot controller was designed, implemented, and validated. The results show that the modular approach ensures the control task with a better precision than a single neural network.

One important advantage of the proposed approach is that the modules are functionally independent and that they can learn without affecting the weights of other modules. Thus, a module is autonomous and, if decomposition is objective, a module can be reused. In our robot application, modules A or C can be reused in order to control the head and to center the target in the images.

Our perspectives concern the development of a more sophisticated active vision scheme which will clearly result in better performances for the tracking task.

References

[1] Jean-Luc Buessler, Jean-Philippe Urban, and Julien Gresser. Additive composition of supervised self-organized maps. *Neural Processing Letters*, 15(1):9–20, 2002.

[2] J.L. Buessler, R. Kara, P. Wira, H. Kihl, and J.P. Urban. Multiple self-organizing maps to facilitate the learning of visuo-motor correlations. In *IEEE International Conference on Systems Man and Cybernetics*, volume 3, pages 470–475, Tokyo, Japan, 1999. IEEE Press.

[3] T. Caelli, L. Guan, and W. Wen. Modularity in neural computing. *Proceedings of the IEEE, Special Issue on Computational Intelligence*, 87(9):1497–1518, 1999.

[4] Masahiko Haruno, Daniel M. Wolpert, and Mitsuo Kawato. Mosaic model for sensorimotor learning and control. *Neural Computation*, 13(10):2201–2220, 2001.

[5] Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

[6] H. Miyamoto, S. Schaal, F. Gandolfo, H. Gomi, Y. Koike, R. Osu, E. Nakano, Y. Wada, and M. Kawato. A kendama learning robot based on bi-directional theory. *Neural Networks*, 9(8):1281–1302, 1996.