A Neuro-Fuzzy Solution for Integrated Visual and Force Control *

Yorck von Collani, Christian Scheering, Jianwei Zhang, Alois Knoll
Technical Computer Science, Faculty of Technology,
University of Bielefeld, 33501 Bielefeld, Germany

Abstract

In this paper the use of a B-spline neuro-fuzzy model for different tasks such as vision-based fine-positioning using uncalibrated cameras and force control is presented. It is shown that neuro-fuzzy controllers can be used not only for low-dimensional problems like force control but also for high-dimensional problems like vision-based sensorimotor control and for fusing input from different sensors. Controllers of this type can be modularly combined to solve a given assembly problem.

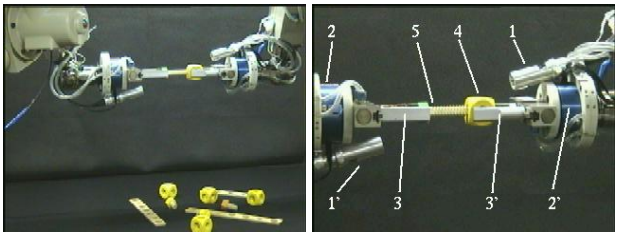
1 Introduction

It is well-known that general fuzzy rule descriptions of systems with a large number of input variables suffer from the problem of the "curse of dimensionality." In many real-world applications it is difficult to identify the decisive input parameters and thus to reduce the number of input variables to the minimum. A general solution to building fuzzy models is not only interesting from a theoretical point, it may also extend the range of applications of fuzzy control to more complex intelligent control problems.

In our research work on sensor-based robot control [7], we are faced with many high-dimensional problems concerning a large number of input variables whose importance and inter-dependence are not clearly known.

1.1 Vision/Force- Guided Robot Motion

a) Assembly skills like inserting and screwing are part of the most important and most demanding sensor-based manipulation skills of cooperating robots. The use of force-feedback is the mostly used sensor information source used in robotics but in recent years visual feedback and especially the integration of both had been of great interest. Conventional techniques try to exploit a common representation space to achieve a fused model of the environment ([9], [3]). In [13] this is achieved by describing sensor observations in terms of uncertain geometry using probabilistic fusion methods. In [1] vision together with an internal strain gauge is used to gather information about the contact forces acting on a hand during grasping. In [12] force and vision feedback is combined using so called *vision and force resolvabilities*. Common to nearly all these approaches is an explicit modeling of the sensor properties in order to combine the information. In our work presented here we integrate visual and information from two uncalibrated cameras with force/torque data obtained from two cooperating arms. We do not use any explicit models but employ an adaptive neuro-fuzzy scheme to *learn* the appropriate robot motions necessary to perform a complex screw-task.



(a) (b)

Figure 1: The experimental setup for assembly. 1,1': hand-camera; 2,2': force/torque sensor; 3,3': parallel jaw-gripper; 4: nut; 5: screw.

b) A special case of vision-action transformation is camera-supported fine-motion control. *Affine Visual Servoing* [4] may be applied to such tasks. The changes in the shape of image contours are the input of a feed-forward controller. In recent years the idea of using uncalibrated cameras for visual guidance has been found increasing interest (c.f.[15, 16, 5, 14]). However, nearly all require some initial and subsequent to perform well.

c) CMAC neural networks may tackle the problem of dimensionality; in [10] 12 inputs represent four joint positions of the robot, four image parameters and their

*The work described in this paper is partly funded by the Deutschen Forschungsgemeinschaft in the project SFB 360/D4.

desired changes. The outputs are the control signals for the four robot joints.

d) In [11] learning of vision-based positioning based on visual appearance information was introduced. The image data set is compressed using *principal component analysis* to obtain a low-dimensional input space. A parametric eigenspace representation is used for describing the different objects as well as object locations. The positioning problem is thus transformed into finding the minimum distance between a point and a manifold in the eigenspace.

e) As far as we know no work on mapping the multiply images direct into "action values" has been reported.

1.2 Existing Solutions to Multivariate Problems

The two main methods to solve the problem of input dimension reduction are *input selection* and *hierarchy*. Input selection [6, 2] is an experimental method to find the most important input variable in a large set of input variables. With this approach, all the combinatorial possibilities of the low-dimensional fuzzy model are considered and approximately tested. The inputs which result in the best outputs are viewed as the most important ones to build an exact neuro-fuzzy model. The problems with this method are the loss of information and the number of combinations that must be tested.

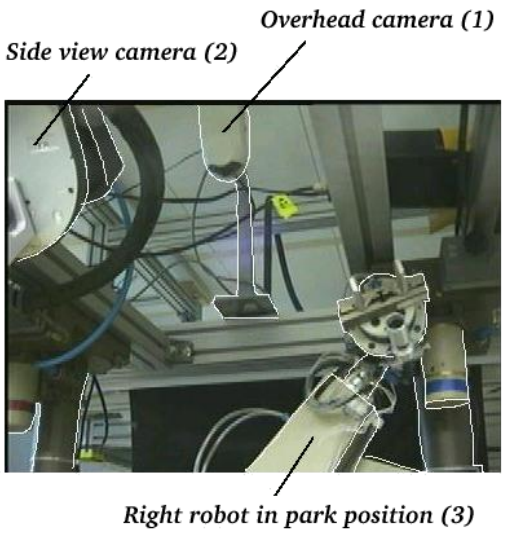


Figure 2: Location of the cameras: (1) overhead, (2) side-view, (3) robot in park position

Hierarchical structuring assumes that the input information can be classified into groups, see [8] for an example. There is, however, no general approach to realize such a grouping.

2 Problem Description

2.1 Experiment Setup

The problem scenario (see Fig. 1 and 2), screwing a screw (5) into a nut (4) with two cooperating robots, originates from our collaborative project which aims at assembly of aggregates with wooden toy construction sets. The manipulators are installed upside down and can grasp the required assembly components from the assembly table. Each robot is equipped with a force sensor (2,2') on which a pneumatic parallel-jaw gripper (3,3') is mounted. A small camera (1,1') is mounted over the gripper. The manipulators are two Puma 260. The host computer is a Sun UltraSPARC. We consider general screwing without using any fixture devices.

2.2 Uncertainties

For a general-purpose arm/gripper system, the following two types of uncertainties must be taken into account:

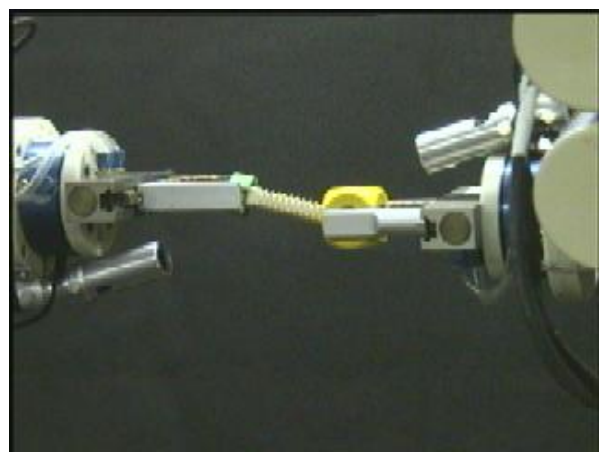


Figure 3: An inconvenient start-situation for screwing

Grasping precision. Although we have applied a hand-camera in a "self-viewing" configuration, which significantly improved the grasping precision in comparison with the open-loop positioning, regrasping still engenders deviation of the screw from the rotation axis of the gripper.

Slippage of the part in the hand. Due to the effect of the resulting forces, the screw grasped by a jaw gripper may easily slip during the screwing process.

The uncertainties in a screwing process cause the following two concrete problems:

1. The screw is not centrically grasped, i.e. the rotating axis of the screw does not match the axis of the gripper.

2. The screw is obliquely grasped, see Fig. 3.

3 Vision-Based Control

Without using sensors a screwing operation can fail under each of the uncertainties discussed above. Therefore, sensor-based compensation motions become necessary. The resulting forces in case 1 in the normal and orientation directions should be minimised and stable. Additionally, to guarantee a successful screwing-in phase, a constant force in the approach direction should be exerted. Unlike the first case, the forces and/or torques give no reliable information about the orientation of the screw. A supplementary approach is to monitor the scene with external cameras and correct the orientation before contact is made between the screw and the nut.

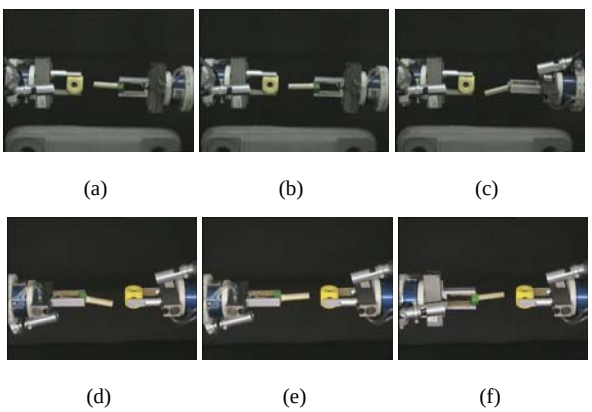


Figure 4: Typical images taken by the external cameras ((a)-(c) viewpoint from above, (d)-(f) side view)

Fig. 4 shows a sequence of typical views of the scene. We therefore employed a method that fuse the visual information from both cameras in real-time (approximately 10 Hz) (in order to deal with 3D uncertainties of the grasped objects), extracts automatically the needed features and combine this information with the "forces" to compensate the uncertainties.

4 B-Spline Neuro-Fuzzy Model

4.1 Basic Principle

The controller for force control can be efficiently realized using the B-spline fuzzy controllers proposed in our earlier work [18]. This type of controller may be characterised by the following features distinguishing it from standard fuzzy controllers:

- B-spline basis functions are employed for specifying the linguistic terms (labels) of the input variables. By choosing the order n of the basis functions, the output is C^{n-2} continuous.
- Each controller output is defined by a set of fuzzy singletons (control vertices). The number of control vertices is equal to the number of the rules and their optimal values can be iteratively found through learning. This adaptation procedure is equivalent to weight adjustment in an Associated Memory Neural Network.
- One problem with learning in conventional fuzzy controllers is that too many parameters must be adjusted. With B-spline fuzzy controllers, a simple modification of control vertices causes the change of the control surface. For supervised learning, if the square error is selected as the quality measure, the partial differential with respect to each control vertex is a convex function. For unsupervised learning, if the error of the cost function is approximately piecewise proportional to the error of the control values, the learning-process descent will also show stable asymptotic behaviour [17].

In [18] we showed the advantages of this learning controller approach. Here, the robot controller learns actively and on-line to control the compensation motion according to measured forces and is quite simple to design. The learning process converges rapidly, the output is smooth if B-spline functions of order 3 or higher are used and the controller performs well.

4.2 Dimension Reduction

If the dimension of the input space is small enough, the input variables can be directly covered by fuzzy sets. Each item of the rule is human readable and may be interpreted as describing a special instance of a general situation. If, however, the image of a camera is regarded as a vector, this high-dimensional sensor image is too large to build a corresponding rule base. Fortunately, sensor images are often observed in a local context: the complete situation is not of particular interest and a subspace can be found that contains all necessary information for determining the action values.

4.3 Projection into Eigenspace

A well-known technique for dealing with multivariate problems in statistics is the *principal component analysis* (PCA). As shown in [11], this technique is also suitable for reducing the dimension of the input space of a general control problem.

Our new append of dealing with 3D uncertainties is to merge small and local parts of different grey-scale camera images and project the resulting image into an eigenspace (see Fig. 5).

(a) (b) (c)

Figure 5: Clipped images from camera 1 (a) and camera 2 (b) and the resulting merged image.

An eigenvector, denoted as EV_i, is computed as $[a_{1,i}, a_{2,i}, \ldots, a_{m,i}]^T$. The eigenvectors form an orthogonal basis for representing the original individual sensor patterns. Assume that the eigenvectors $EV_1, EV_2, \ldots$ are sorted according to their eigenvalues in a descending order. An eigenspace with a reduced dimension n can be formed with the first n eigenvectors. EV_i defines the ith dimension in the eigenspace. The projection of an input vector $X = [x_1, x_2, \cdots, x_m]^T$ onto eigenvector EV_i, called the ith principal component, is $p_i = a_{1,i}x_1 + a_{2,i}x_2 + \cdots + a_{m,i}x_m$. The complete projection can be represented as:

$$[EV_1, \ldots, EV_n]^T \cdot X = [p_1, \ldots, p_n]^T$$

All projections of the sample data sequence form a manifold in the eigenspace. Such a projection can be viewed as a layer of neural network, see the connection layer of the two left parts of Fig. 6.

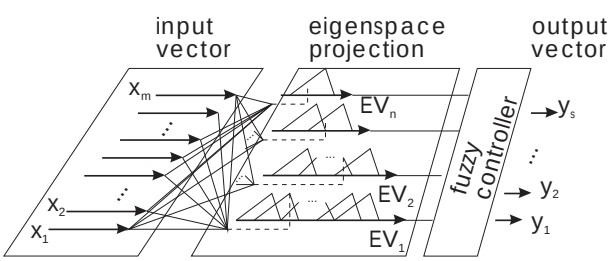


Figure 6: The structure of a fuzzy controller based on eigenspace projection.

We partition the eigenvector with the largest eigenvalue with the finest resolution because the projection on this eigenvector contains the largest variance. If the eigenvectors $\mathrm{EV}_1, \ldots, \mathrm{EV}_n$ are sorted in a descending order according to their eigenvalues, then EV_1 is the first eigenvector. The second eigenvector EV_2 is covered with fewer linguistic terms, and so on.

5 Implementation

5.1 Sampling Training Data

For training, the input data and desired output values have to be recorded. It is desirable that all typical input data be generated. As outlined above there are different orientations of the screw. For recording, the robot moves to the ideal position and orientation for screwing. Subsequently, it moves to several other orientations. For each of them the deviations from the ideal orientation are recorded. The screw is regrasped several times to get data scattering.

5.2 Calculating Eigenvectors

After the input data are sampled, the following steps are necessary:

1. The input data are normalised so that the energy of each image becomes 1. As an option the average image can be subtracted.
2. The input variables are stacked into vectors.
3. The covariance matrix of the input vectors is calculated.
4. The eigenvectors and eigenvalues are calculated.
5. Each image is projected into the eigenspace.

5.3 Training the Fuzzy Controller

For the B-spline controller the training procedure is as follows:

1. Select the n eigenvectors with the largest n eigenvalues denoted as $\mathrm{EV}_1, \ldots, \mathrm{EV}_n$.
2. Select the order of the B-spline basis function for each eigenvector.
3. Determine the knots of the B-spline basis functions for partitioning each eigenvector.
4. Project images onto the selected eigenvectors.
5. Initialise the control vertices for the output.
6. Learn the control vertices with the projected values from the images using the gradient descent method.
7. If the results are satisfying, terminate.
8. Modify the knots for eigenvectors, go to 5.

It is important to determine the right parameters for the fuzzy set. If too few eigenvectors are used, then the fuzzy controller cannot distinguish all situations. If too many eigenvectors are used, then the memory requirements of the fuzzy set is not manageable. Similarly important is the correct partition. If the partitioning is too fine, the fuzzy controller generalises insufficiently.

6 Numerical Results

Fig. 7 shows the sorted eigenvalues of the covariance-matrix and Fig. 8 the resulting control surface for the first two eigenvectors. The vision controller is learned with 391 training images, shifting the screw between $\pm 15°$ around the N- and O-direction. The learned controller is tested with additional 191 images. Fig. 8 shows clearly that more than two eigenvectors are needed to detect the right orientation of the screw. We use the eight greatest eigenvectors

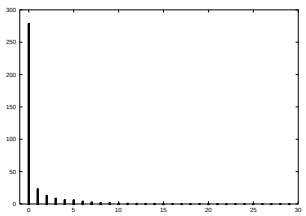


Figure 7: Sorted eigenvalues of covariance-matrix.

and cover the dimensions with $10 \times 9 \times 8 \times 7 \times 6 \times 5 \times 4 \times 3$ B-Splines as membership functions.

For comparison Table 1 shows the mean square error of the controller, the maximum error and the largest worst case errors of the test images around the O-direction. The worst case is defined as a correction in the wrong direction. Our experiment shows that merging the images pro-

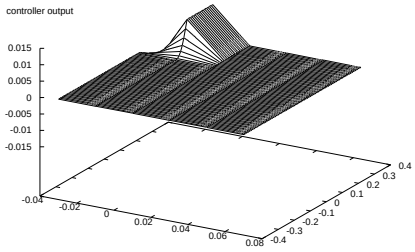


Figure 8: Control surface with the first two eigenvalues as input and the controller output.

duce a much better controller than the separated controller for each camera and direction. The mean square error and

	Mean square error $[\circ^2]$	Max. error $[\circ]$	Worst case error $[\circ]$
Camera 1	114.18	24.13	24.13
Camera 2	134.18	25.02	25.02
Fused images	59.23	15.0	8.37346

Table 1: Mean square error, maximum error and worst case error for angle around O-direction.

maximum error are significantly smaller. In case of using only one image the worst case errors are to high to be compensated by the controller. With the merged images in contrast the worst case error is significantly smaller (eight degree) and is successfully corrected. Fig. 9 shows the required correction steps around N- and O-direction for different orientations. Apart from a few exceptions only one to three steps are required to correct the orientation.

Combining this approach with the first procedure described in [18] results in a very robust and rapid technique to determine the orientation correction of the screw.

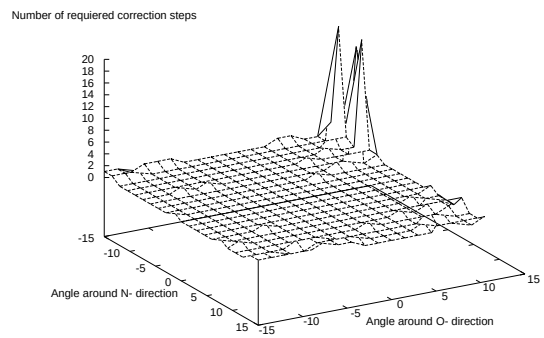


Figure 9: Required correction steps for different orientations of the screw.

7 Conclusions

We have shown that high-dimensional problems such as visually guided fine-motion can be solved with neuro-fuzzy controllers. We have also shown that the B-spline model may be utilised for sensor fusion and for different problems, not only for force control. We have implemented the approach with a two-arm robot system and both kinds of training are used to build the controllers: unsupervised online for the force controller and supervised off-line learning for the vision system.

The advantages of our approach are:

- By merging the different views into one image and projecting the high-dimensional input space into a reduced eigenspace, the most significant information for control is maintained. A limited number of transformed inputs can be partitioned with the B-spline model and a sufficient precision can be obtained for determining the robots orientation correction.
- To solve this problem the statistical indices provide a suitable solution to describe the information in images with a lot of uncertainties.
- A vector in the eigenspace is directly mapped onto the controller output based on the B-spline model. This makes real-time computation possible.
- Designing the controllers is simple and identical for both low and high dimensional controllers. Both force and vision controllers are of the same type. The B-spline fuzzy controller can be trained in a straightforward manner because modification of control vertices only results in local change of the control surface.

In this approach no complex programming and knowledge about vision is needed. We have shown that this approach is very promising for realizing efficient robot assembly skills based on sensorimotor coordinations.

References

[1] P. K. Allen, A. T. Miller, P. Y. Oh, and B. S. Leibowitz. Integration of visison and force sensors for grasping. In *Proc. IEEE Conf. on Multisensor Fusion and Integration for Intelligent Systems*, pages 349–, 1996.

[2] S. L. Chiu. Selecting input variables for fuzzy models. *Journal of Intelligent and Fuzzy Systems*, 4:243–256, 1996.

[3] J. L. Clark and A. L. Yuille. *Data Fusion for Sensory Information Processing Systems*. Kluwer Academic Publishers, 1990.

[4] C. Colombo, B. Allotta, and P. Dario. Affine visual servoing: A framework for relative positioning with a robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 464–471, 1995.

[5] M. Jägersand, O. Fuentes, and R. Nelson. Experimental evaluation of uncalibrated visual servoing for precision manipulation. In *Proc. IEEE Int. Conf. Robot. Automat.*, pages 2874–, 1997.

[6] J. S. R. Jang, C. T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing*. Prentice Hall, 1997.

[7] A. Knoll, B. Hildebrandt, and J. Zhang. Instructing cooperating assembly robots through situated dialogues in natural language. In *IEEE International Conference on Robotics and Automation*, 1997.

[8] V. Lacrose and A. Tilti. Fusion and hierarchy can help fuzzy logic controller designers. In *IEEE International Conference on Fuzzy Systems*, 1997.

[9] G. T. McKee. What can be fused? In J. K. Aggarwal, editor, *Multisensor Fusion for Computer Vision*, NATO ASI Series, pages 71–. Springer-Verlag, 1993.

[10] W. T. Miller. Real-time application of neural networks for sensor-based control of robots with vision. *IEEE Transactions on System, Man and Cybernetics*, 19:825–831, 1989.

[11] S. K. Nayar, H. Murase, and S. A. Nene. Learning, positioning, and tracking visual appearance. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3237–3244, 1994.

[12] B. J. Nelson and P. K. Khosla. Force and vision resolvability for assimilating disparate sensory feedback. *Transaction on Robotics and Automation*, 12(5):714–, 1996.

[13] Hugh F. Dur rant Whyte. *Integration, Coordination and Control of Multi-Sensor Robot Systems*. Kluwer Academic Publishers, 1988.

[14] C. Scheering and B. Kersting. Uncalibrated hand-eye coordination with a redundant camera system. In *Proc. ICRA-98*, 1998.

[15] S. Skaar, W. Brockman, and W. Jang. Camera-space manipulation. *Journal of Robotics Research*, 6(4):20–32, 1987.

[16] B. H. Yoshimi and P. K. Allen. Active, uncalibrated visual servoing. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 156–161, 1994.

[17] J. Zhang and A. Knoll. Constructing fuzzy controllers with B-spline models – principles and applications. *International Journal of Intelligent Systems*, 13(2/3):257–286, February/March 1998.

[18] J. Zhang, Y. v. Collani, and A. Knoll. B-spline fuzzy controller to acquire sensor-based assembly skills. In *IEEE International Conference on Robotics and Automation*, 1997.