

Fast Object Recognition for Grasping Tasks using Industrial Robots

Ismael López-Juárez¹, Reyes Rios-Cabrera¹, Mario Peña-Cabrera²,
Gerardo Maximiliano Méndez³, and Román Osorio²

¹Centro de Investigación y de Estudios Avanzados del IPN (CINVESTAV),
Mexico

²IIMAS-UNAM,
Mexico

³Instituto Tecnológico de Nuevo León (ITNL),
Mexico

ismael.lopez@cinvestav.edu.mx, reyes_rios@hotmail.com, mario@leibniz.iimas.unam.mx,
gmm_paper@yahoo.com.mx, roman@unam.mx

Abstract. Working in unstructured assembly robotic environments, i.e. with unknown part location; the robot has to accurately not only to locate the part, but also to recognize it in readiness for grasping. The aim of this research is to develop a fast and robust approach to accomplish this task. We propose an approach to aid the learning of assembly parts on-line. The approach which is based on ANN and a reduced set of recurrent training patterns which speed up the recognition task compared with our previous work is introduced. Experimental learning results using a fast camera are presented. Some simple parts (i.e. circular, squared and radiused-square) were used for comparing different connectionist models (Backpropagation, Perceptron and FuzzyARTMAP) and to select the appropriate model. Later during experiments, complex figures were learned using the chosen FuzzyARTMAP algorithm showing a 93.8% overall efficiency and 100% recognition rate. Recognition times were lower than 1 ms, which clearly indicates the suitability of the approach to be implemented in real-world operations.

Keywords. Artificial neural networks, invariant object recognition, machine vision, robotics.

Reconocimiento rápido de objetos para tareas de agarre usando robots industriales

Resumen. En celdas de ensamble robotizado en ambientes no estructurados, por ejemplo con localización de partes desconocidas, el robot tiene, no solamente que localizar la parte, sino también reconocerla para su agarre. El objetivo de esta investigación es desarrollar un enfoque rápido y

robusto para completar la tarea. El enfoque basado en RNA y un reducido conjunto de patrones recurrentes de entrenamiento que aumentan la tarea de reconocimiento comparado con nuestro trabajo es introducido. Se presentan los resultados de aprendizaje experimental utilizando una cámara rápida. Algunas partes simples (es decir, circulares, cuadrados y semi-cuadrado) fueron utilizados para comparar diferentes modelos conexionistas (Backpropagation, Perceptrón y FuzzyARTMAP) y para seleccionar el modelo apropiado. Más tarde, durante los experimentos, se aprendieron figuras complejas mediante el algoritmo de FuzzyARTMAP elegido mostrando un 93,8% tasa de reconocimiento global de eficiencia y un 100% en la razón de reconocimiento. Los tiempos de reconocimiento fueron inferiores a 1 ms, lo que indica claramente la idoneidad del enfoque para implementarse en operaciones de mundo real.

Palabras clave. Redes neuronales artificiales, reconocimiento invariante de objetos, visión de máquina, robótica.

1 Introduction

Grasping and assembly operations using industrial robots is currently based on the accuracy of the robot and the precise knowledge of the environment, i.e. information about the geometry of assembly parts and their localization in the workspace. Techniques are sought to provide self-adaptation in robots. This document reports a neural-based methodology for invariant object recognition applied to self-adapting

industrial robots which can perform assembly tasks. New objects can also be learned quickly if certain clues are given to the learner, since the methodology proposed here compared to our previous approach [1], uses only two on-line patterns for learning complex objects. The architecture is firstly trained with clues representing different objects that the robot is likely to encounter (and with others that represent complex objects) within the working space to form its initial knowledge base. The main idea suggests that it is possible to get fast and reliable information from a simple but focused analysis of what an object might show. The very important aspects of the scene (we have called “clues”), can be used later to retrieve memorized aspects of the object without having to recall detailed features. In some way, humans do that process once an object has been seen and learned for the first time.

The article describes a robust method for very fast learning using ANN’s, perimeter and centroid calculations, object functions and pose estimation. The proposal is centred on very few training patterns (two) which trigger network learning using recurrent patterns during training. In order to validate our algorithm several test were carried out using simple and complex figures that were learned using the chosen FuzzyARTMAP algorithm. The network showed a 93.8% overall efficiency and 100% recognition rate. Recognition times were lower than 1 ms, which clearly indicates the suitability of the approach to be implemented in real-world operations with industrial robots as demonstrated in previous work [2].

The remainder of this paper is structured as follows. Section 2 reviews related work and state our contribution to the field of self-adaptive industrial robots for assembly and object recognition. In Section 3, the analysis of three different ANN’s is presented for selecting the appropriate connectionist model while in section 4 the methodology is explained. Experimental results from several object learning and recognition tasks are given in section 5. Finally, conclusions and future work are described in section 6.

2 Background Work Domain

The robot work domain is formed basically by a 6 DOF KUKA KR16 industrial robot, KRC2 robot controller, KUKA Control Panel (KCP), PC Master Computer, JR3 F/T sensor attached to the robot’s wrist, a ceiling mounted CCD camera and a conveyor belt as it is illustrated in Figure 1. The main units of the robot system are the KRC2 controller and the robot arm itself. The KRC2 controller houses the components that control and power the robot arm. The Master Computer host the DSP-based F/T sensor card and also communicates with the robot controller at lower level via serial port. The vision system uses the master computer in which algorithms for POSE determination (orientation and location) reside. POSE information about the components on the conveyor belt in turn issues proper motion commands to the KRC2 controller for component grasping. Once the part (male component) is held by the robot, then the vision system also determines the female location at the Master Assembly Block and sends the female centroid information to the Master Computer in order to move the male component above the female component in readiness for assembly.

The focus in this paper is on the object recognition system instead of the assembly task. Figure 2 shows the assembly components used for recognition and learning of manipulative tasks (i.e. assembly operations). These components have different cross-sectional form (circular, squared and the so-called radiused-square).

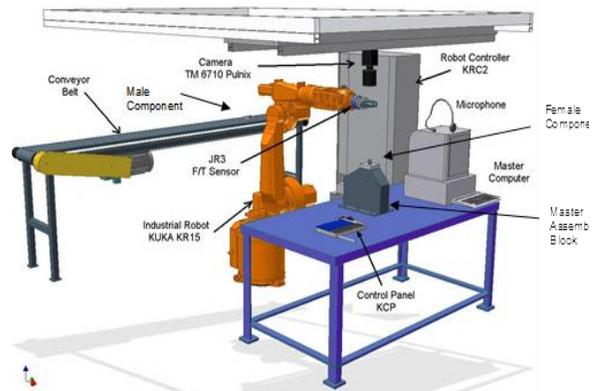


Fig. 1 Assembly Cell

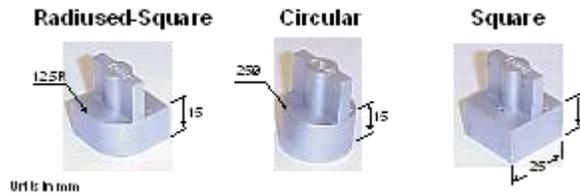


Fig. 2. Assembly Components

2.1. Related Work

Many authors have considered only constraint motion control during assembly; however, to complete the autonomy of the assembly system a machine vision system has also to be considered. Hoska introduced the concept of “Robotic Fixtureless Assembly” (RFA) that eliminates the need of using complex and rigid fixtures, which involves new technical challenges, but allows potential solutions [3]. Ngyyuen and Mills [4] have studied RFA of flexible parts with a dynamic model of two robots with an algorithm that does not require measurements of the part deflections. The goal of RFA is to replace these fixtures with sensor-guided robots which can work within RFA workcells. By using ANNs, an integrated intelligent vision-guided system can be achieved [5]. Many authors have used descriptor vectors and image transformations as general methods for computer vision applications in order to extract invariant features from shapes. Aguado, developed a methodology for including invariance in general form of the Hough transform [6], Chin-Hsiung *et al* [7] designed a technique for computing shape moments based on the quadtree representation of images. Best and McKay described a method for registration of 3D shapes in minutes [8]. Some authors use multiple cameras/views to extract information. For invariant object recognition and to determine the object’s position and motion, Gonzalez-Galvan *et al* [10] developed a procedure for precision measure in 3D rigid-body positioning using camera-space manipulation for robot operation [10]. Applications of guided vision used for assembly are well illustrated by Bone and Capson, which developed a vision-guide fixtureless assembly system using a 2D computer

vision for robust grasping and a 3D computer vision to align parts prior to mating [11].

In our previous method it was reported a 100% object recognition rate for the objects showed in figure 2. In this case several patterns were used for training each of the three objects. (The reader is referred to [1] for details). The vision system was composed by a CCD camera with 640x480 pixel resolution. POSE information was provided by the vision system to the master computer to generate the robot motion commands for grasping the right component for assembly. In figure 3 the used procedure is illustrated. The robot itself served as positioning device during training stage in order to generate different object images at different scale, location and orientation. With this approach a total of 72 descriptive vectors were needed during training for each component.



Fig. 3. Multiple view object training

2.2. Original Work

The research presented in this paper is focused on learning to recognize simple and complex 2D objects in order to enable industrial robots to learn manipulative tasks (i.e. assembly operation). In this area, moment invariants are still popular descriptors for image regions and boundary

segments, however, computation of moments of a 2D image involves a significant amount of multiplications and additions in a direct method. Some of the methods require multiple-pattern input with considerable computing overload which is not appropriate for real-world applications with industrial robots, that is, the robot requires recognizing quickly (approx. 1 sec) the part, to be grasped in order to continue with the following task.

In this paper we propose the recurrent use of pattern vector descriptors, using collections of 2D images to obtain a very fast feature data of an object by using image projections and mirror images. Fast learning is achieved on-line considering only two patterns of simple (geometrical) and complex objects to achieve invariance to rotation, translation and scale. Our intention is to implement an approach similar to the human object recognition in which a fast learning can be achieved in one shot and the objects can easily be recognized. Compared to our previous work in which we had to train multiple patterns, we propose the following hypothesis: **“With only two patterns from a regular or irregular object from a similar viewpoint, it is possible to learn and recognize the object on-line achieving invariance to: rotation, location and scaling for a 2D image representation of a 3D object”**.

The basic idea is to emulate the human learning/recognition ability, since humans need only to observe the object once or twice in order to remember it later.

The fast algorithm allows calculation of a boundary object function (BOF) and centroid which defines object information, and also considers variance normalized grey-color intensity properties, which forms a unique vector descriptor that in conjunction with an ANN recognizes, learns and performs pose estimation of assembly components in the order of milliseconds what constitutes a practical tool for real-world robot applications.

3 Neural Networks Evaluation

An experimental comparison among BP (Backpropagation), P (Perceptron), and FAM

(Fuzzy ARTMAP) was made in order to select an appropriate model for the vision system. A particular set for training and testing was selected based on our basic assembly components showed in figure 2 and whose pattern shape is as illustrated in figure 4. These pattern sets were considered for benchmark purpose in the ANN's considered in this study and to assess its convergence learning time.

The pattern is formed by the contour of the respective piece and provided by the Boundary Object Function (BOF) after image processing. The BOF is a function that describes a specific shape (distances perimeter-centroid) as given in eq. (1)

$$BOF = [D_1, D_2, D_3, D_n, X_c, Y_c, \phi, Z, ID]^T \quad (1)$$

In this equation BOF is a descriptive vector where:

D_i are distances centroid-perimeter,

X_c, Y_c centroids,

Φ orientation,

Z the object's height,

ID is a code number related to the geometry of the component.

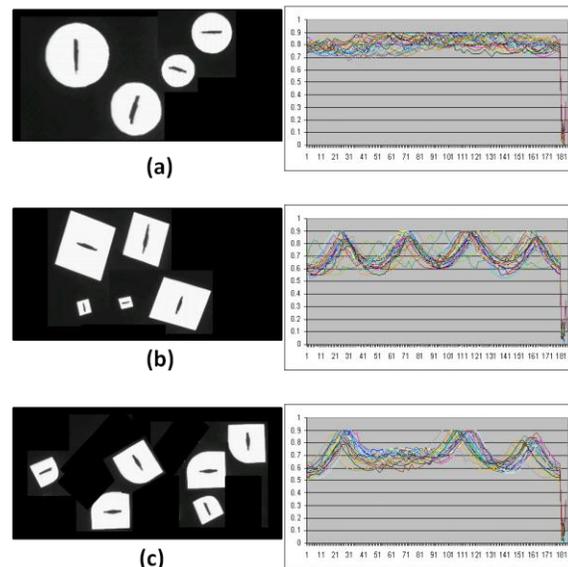


Fig. 4. a) circular, b) square, c) radiused-square. Total = 216 patterns (72 from each shape)

Five experiments were made with all the ANN's under evaluation. Their algorithm was programmed in Visual C++ .NET using a Pentium D PC @ 2.8 GHz with 512 MB RAM.

3.1 Backpropagation

BP is a stochastic steepest descent learning rule used to train single or multiple layer nonlinear networks. The algorithm overcomes some limitations of the perceptron rule by providing a framework for computing the weights of hidden layer neurons, but also it takes more time for training/testing. The configuration was: layer_in=185, layer_hidden=200, layer out=4, the weights were selected layer 1, 2: Random weight [-2.0, 2.0], and layer 3, Random weight [-1.0, 1.0]. The learning rate $\alpha=0.7$ and maximum error allowed was 0.12 (the employed patterns were from showed objects in figure 2). Other experiments were made using, topology 185-300-4, learn rate=0.85, 185-250-4 showing less efficiency.

Since this ANN depends on randomly selected weights, five experiments were made, and the average of the experiments is considered for comparison purposes. The figure 5 shows the performance of the ANN in learning. Since there were only 3 objects to recognize, the classification starts with a 64.16% error (the error means the % of patterns that are not recognized from the universe of 216). The best case in 675 epochs reached 0% error in 57.593 s. (training), in the worst case 1332 epochs in 116.375 s.

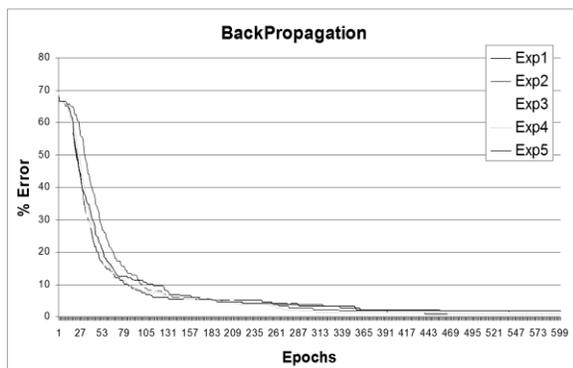


Fig. 5. a) Backpropagation performance

3.2 Perceptron

It is a feedforward network with one or more outputs that learn the position of a separating hyperplane in pattern space (a layer for nonlinearly separable pattern pairs). The first layer has fixed weights and the second change according to the output's error. If there is no such error, then the neuron's weights are not modified. This architecture was considered, because it can reach always a no-linear pattern classification properly with enough number of neurons in layer 1. In [12] it was demonstrated that one hidden layer is sufficient to perform any arbitrary transformation using enough nodes. This model was considered for comparison purposes due to its high training/testing speed compared to BP.

The used configuration during five experiments employed 185 inputs, 4 outputs, 450 neurons in Layer1: 450 neurons (10 C/N=Connections per neuron), Random weights [-2.0, 2.0], Layer2: 4 neurons, Threshold: 0.0 (both layers, signum threshold device), $\alpha=0.85$. Other experiments were done using 450 (6, 8, 10, 12 C/N)-4, 350 (6, 8, 10, 12 C/N)-4, showing very similar results. The network's error behaviour at the beginning shows an average error of 63.8% (see figure 6). This ANN showed a much better performance than the BP, in the best case reaching 0% error in 41 epochs and a training time: 0.8112s, the worst case took 83 epochs, with 1.734s training.

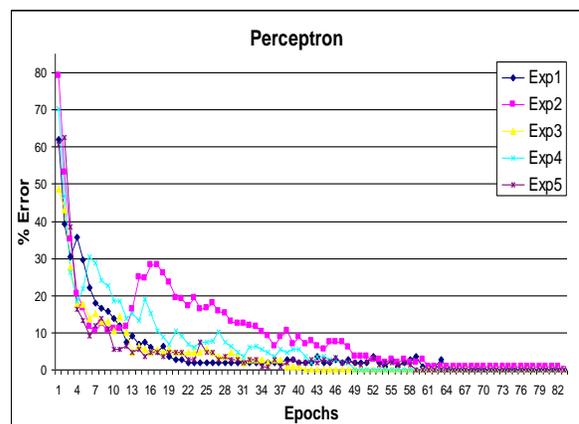


Fig. 6. Perceptron performance

3.3 Fuzzy ARTMAP (FAM)

The Fuzzy ARTMAP network was developed by Carpenter and Grossberg based on the ART (Adaptive Resonance Theory) in which supervised learning is carried out. This ANN creates several neurons according to the number of patterns and the differences among them. It has several advantages compared with traditional neural networks such as Backpropagation, since it does not suffer catastrophic forgetting [13]. The configuration for this architecture was, 2 epochs, $\rho_{map} = 0.8$; $\beta = 1.0$; $\rho_a = 0.7$; $\rho_b = 1.0$; aF1Size = 185; bF1Size = 4. For all experiments α was set to 0.1. Five experiments were carried out. This ANN does not depend on random values and that is the reason why the same value was obtained in the graph Error vs. Epochs.

This ANN starts from 100% error reaching 0% error in one epoch. The best training time was 0.172 s and the worst 0.188s. See figure 7. In all cases, the generated Knowledge Base (internal representation of the network) showed the same behavior: For the circle one neuron was generated, and for the square and radiused-square, 2 neurons. This is because it was configured to a maximum data compression.

Figure 8, is a 100 epochs experiment with the same patterns, varying parameters β , ρ_a , α and ρ_{map} . This experiment was done for testing the stability of the network and for selecting the best network parameters. The graphic shows that the network continues creating new neurons along the epochs with less variation at the end of the graph. It was noticed that β and ρ_a mostly determined the stability in the network.

The property of FAM encoding critical features is a key to code stability. This learning strategy makes the difference of ART networks and MLP's (Multilayer Perceptrons), which typically encode the current input, rather than a matched pattern, and hence, employ slow learning across many input trials to avoid catastrophic forgetting [14].

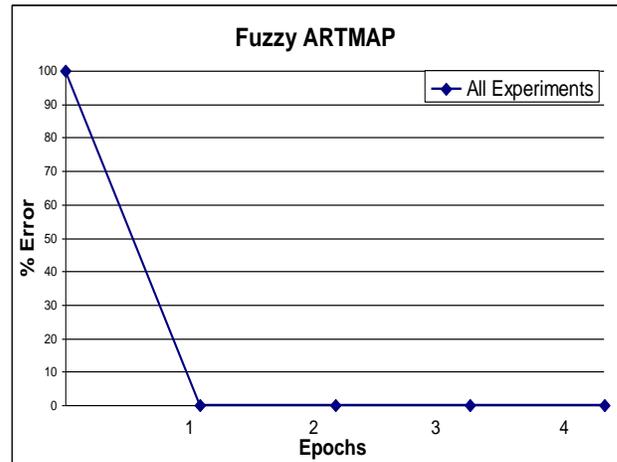


Fig. 7. FuzzyARTMAP error

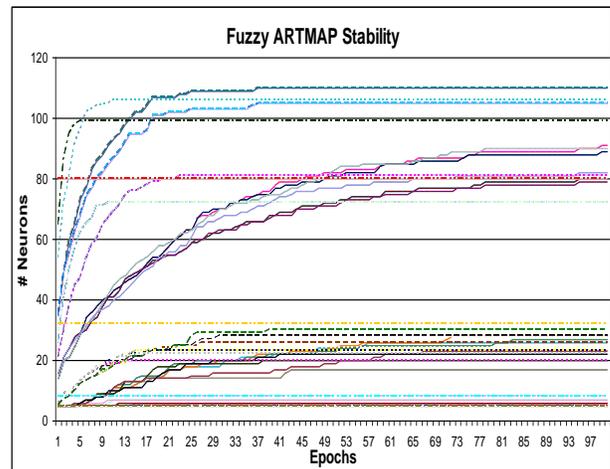


Fig. 8. FuzzyARTMAP Stability-sensibility parameters

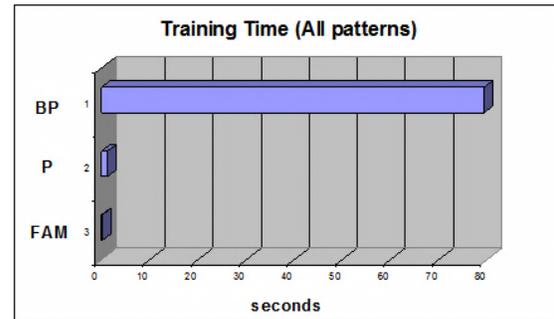
3.4 Evaluation Results

Table 1 shows the results of the experiments. Time average is showed in figure 9. With these results the FAM was selected because of its incremental knowledge capabilities and stability, but mostly because of the fast recognition and geometrical classification responses.

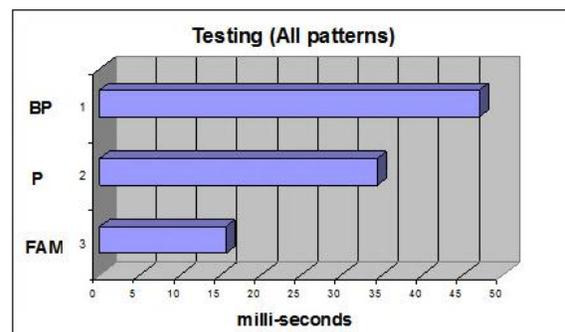
Table 1. Results of the ANN experiments. Time is given in seconds and training/testing all patterns

Training time (s)			
No.	Backpropagation	Perceptron	FuzzyARTMAP
1	79.625	1.391	0.188
2	116.375	1.734	0.187
3	68.734	0.8112	0.187
4	57.593	1.109	0.172
5	74.657	1.203	0.172
Average	79.39	1.24	0.18

Testing time (s)			
No.	Backpropagation	Perceptron	FuzzyARTMAP
1	0.047	0.031	0.015
2	0.047	0.031	0.015
3	0.047	0.016	0.016
4	0.047	0.047	0.016
5	0.047	0.047	0.016
Average	0.047	0.034	0.015



(a)



(b)

Fig. 9. a) ANN's Training time - b) ANN's Testing time. BP showed a Train-Test time of 367.577 ms - 0.217 ms per pattern, the P, 5.78 ms - 0.159 ms per pattern, and FAM 0.838 ms - 0.0722 ms per pattern

4 Object Recognition Methodology

4.1 Previous Approach

As previously mentioned the Boundary Object Function (BOF) is the function that describes a specific shape given by eq. (1). This method was tested with 3 shapes for on-line recognition, the results showed training/testing time of few ms and a 100% classification. In practice we have tested the algorithm for recognition and robot grasping tasks; however, the methodology needed several patterns for training, then adding a new object would not be fast for real world operations. The hypothesis of the new approach is that with only two patterns from a regular or irregular object, it is possible to learn and recognize the object on-line achieving invariance to: rotation, location and scaling of a 2D image representation of an object.

4.2 Pattern Generation: Proposed approach

In order to subtract only parameters from the object, the proposal is to take the BOF and grey-scale characteristics: the average grey color and the object-histogram.

We suppose a robotic fixtureless scenario in which there is no specific guide or line to find the orientation of a piece. We can approach a line through the piece and determine the orientation angle, but we can obtain only a value 0-180° (in both cases 0° to 180° or 180° to 360°).

This behavior will generate two different patterns for the same object that cannot be rotated and rearranged automatically for the lack of the real angle.

In the conveyor belt, it is also possible for the robot to find an object that is the mirror image of

the trained one (the other object's side). In order to classify those mirror images also in the ANN, it is necessary to reproduce those in the inverse order during the training stage in such a way to have a new pattern to classify (2 on-line images and 2 mirror images) as illustrated in figure 10.

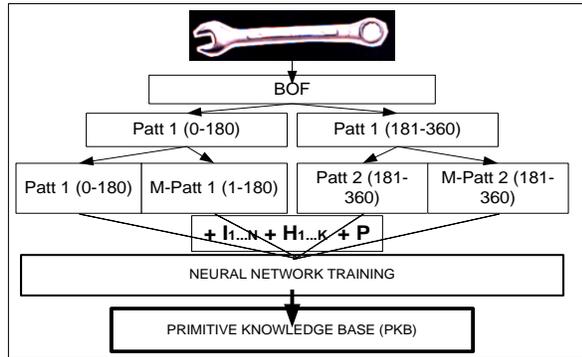


Fig. 10. Knowledge base generation using two simple patterns

With these initial patterns, it is possible now to rearrange all other not trained patterns with a rotation function based on the founded orientation and generating very similar patterns. For managing the size invariance, normalization is applied. The grey-scale intensity average is an important characteristic of an object, and it is included in the object-properties vector as well as its histogram.

The histogram no matter the size, orientation or light conditions (within a working range), has implicit information about the object and behaves similarly. It is proposed in this work the descriptive vector given by a Generalized Boundary Object Function (BOFG) as given in eq. (2)

$$BOF_G = [BOF + I_{1..N} + H_{1..K} + P] \quad (2)$$

where:

BOF_G: Generalised Boundary Object Function

BOF: Boundary Object Function

I: Grey-scale intensity average copied *n* times

H: object-histogram considering *K* points

P: any other invariant property.

The figure 11 shows the generation of a BOF_G vector with the properties of a wrench. The descriptor vector uses 180 BOF data, 10 Intensity

data, and 30 data coming from function histogram, all data is normalized to the range[0,1].

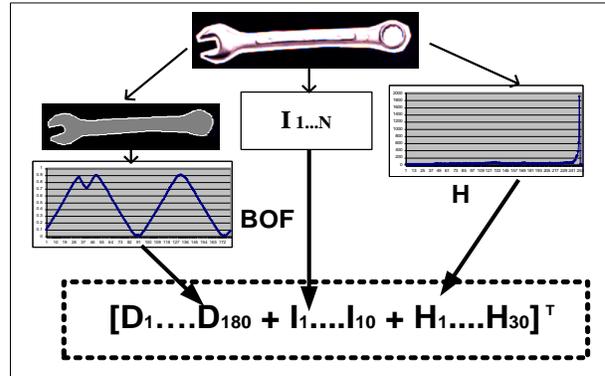


Fig. 11. BOF_G Descriptor vector generation

In other words a pattern such as the one presented in figure 12 will be used as input vector to the FAM network. For the first experiments, the whole pattern sets were made of [BOF + I], and the histogram was not used in order to test part of the vector only. Later experiments where done, using the H (BOF_G), for the group of similar figures.

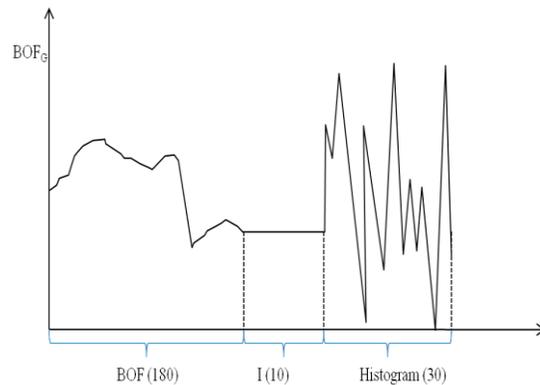


Fig. 12. BOF_G Descriptor vector

In order to test the performance of our algorithm, a set of complex figures in 2D were used to train the FAM network. Several complex figures from animals and tools were used as well as simple geometrical figures. Figure 13 shows the universe of 20 images used for the experiments. Each figure had 3 different sizes so

making a universe training set of 60 images. In figure 14 an example from the BOF from 5 animal figures is presented.



Fig. 13. 20 different figures: small, medium and large (total = 60 Patterns)

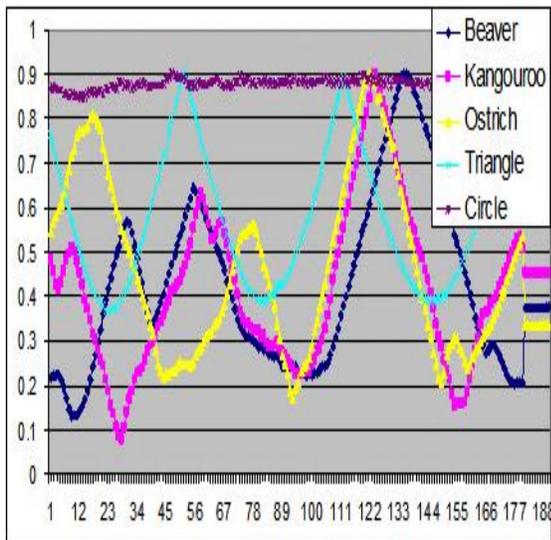


Fig. 14. Patterns for 5 figures

One of the advantages of the algorithm is that while finding the perimeter and marking the object, the centroid, mass, histogram, etc., are calculated at the same time. Once this is completed, then the descriptor vectors are generated. All data is normalized to the range [0, 1], and the number of points fixed to vector size.

5 Experiments and Results

The algorithms were coded in Visual C++ .NET using a Pentium D PC @ 2.8 GHz with 512 MB RAM. For taking the images in real-time, a high speed, 100 fps, IEEE 1394 Basler f602c 640x480 resolution camera was used. From the universe of 20 different objects (three sizes for each one) as observed in figure 13, 20 different middle size were selected for training purposes, placing each object in two different angles (values within 1° to 180° and 181° to 360° range). Two patterns were generated, as well as their mirror images (40 images were captured on-line and 40 mirror images created, for the 20 different figures) for creating the knowledge base in the ANN. For the first experiment only, the [BOF + I] was used in order to evaluate the performance. After training, the experiments consisted of testing the 60 pieces, even though they were not trained, with different location, and orientation. For testing: Size 1, 2, 3, Angle 45, 135, 225, 315, Sides: 1, 2, Total=480 patterns were used. The results are given in Table 2.

Most of the errors in table 2 were detected with animals with similar bodies, (Tiger, Bear, Zebra, Elephant, etc.). The solution to this problem could be adding more patterns, but then the hypothesis will not be fulfilled. The other option is to add the object histogram to complete the descriptor vector.

For this group of animals, the complete vector was generated (BOF_G), new experiments were done and the results were successfully achieved. From these results we concluded that if we have very similar objects it is necessary to use the generalized vector (BOF_G) and for simple or medium complex figures with the basic form is enough. We can observe that the ANN created only one neuron for recognizing all patterns of its group for objects 14 and 15, because of the

Table 2. For geometric pieces 100% recognition was achieved. An overall average recognition of 93.8% was achieved. 6 out of 20 objects were not 100% (similar objects). If we consider only the No Mirror part (side 1) for testing, the achievement is 97.8%. Training time was 1.95 ms per pattern and 0.2 ms for testing. FAM parameters were: 2 epochs, $\rho_{\text{map}}=0.92$; $\beta=1.0$; $\rho_a=0.9$; $\rho_b=1.0$. Err=Error, T=total, NM= No Mirror, N=Neurons

No	Object	Error side 1	Error side 2	Total Error	% OK	%OK NM	Added Neuron
1	Porcupine	0	0	0	100	100	4
2	Beaver	0	0	0	100	100	4
3	Kangaroo	0	0	0	100	100	4
4	Ostrich	0	0	0	100	100	4
5	Giraffe	1	3	4	83.3	91.7	4
6	Elephant	2	4	6	75	83.3	4
7	Hare	0	1	1	95.8	100	4
8	Zebra	1	3	4	83.3	91.7	4
9	Tiger	1	2	3	87.5	91.7	4
10	Rhinoceros	2	4	6	75	83.3	4
11	Raccoon	0	0	0	100	100	4
12	Bear	2	4	6	100	100	4
13	Triangle	0	0	0	100	100	3
14	Hexagon	0	0	0	100	100	1
15	Circle	0	0	0	100	100	1
16	Parabola	0	0	0	100	100	2
17	Hammer	0	0	0	100	100	4
18	Drill	0	0	0	100	100	3
19	Brush	0	0	0	100	100	4
20	Wrench	0	0	0	100	100	3
	TOTAL			30	94.90	97.08	69

object simplicity. 69 neurons were created for recognizing the universe of 20 objects, 3 sizes, all angles.

For sending information to the robot once in the manufacturing cell, a secondary vector is considered, (based on the recognized object and a data base related to it) in eq. (3), where Cx, Cy is the POSE of the object, Φ is the orientation angle, ID is the identified object, and OI is Object Information related to the object grasping taken from the data base.

$$[Cx \ Cy \ \Phi \ ID \ OI] \quad (3)$$

6 Conclusions and Future Work

An invariant method using 2D image object representation suitable for part grasping during robot operations was presented. The method uses only two image patterns from a regular or irregular object instead of using multiple patterns for training. From the given results it was demonstrated that it is possible to learn and recognise the objects on-line achieving invariance to: rotation, location and scaling.

The method showed potential fast recognition and accuracy in the classification of simple and complex 2D objects. The superiority of FAM for this task was also experimentally demonstrated

compared to other connectionist models. Recognition times during testing can be lower than 1ms for practical purposes which is appropriate for real-time when working with industrial robots.

For future work it is intended to prove the algorithms in the robotic testbed using varying light conditions.

Acknowledgements

The authors wish to thank CONACyT through project research grant No. 61373-Y.

References

1. Peña-Cabrera, M., Lopez-Juarez, I., Rios-Cabrera, R., & Corona-Castuera, J. (2005). Machine vision approach for robotic assembly. *Assembly Automation*, 25(3), 204–216.
2. Corona-Castuera, J. & Lopez-Juarez, I. (2006). Behaviour-based approach for skill acquisition during assembly operations, starting from scratch. *Robotica*, 24(6), 657–671.
3. Hoska DR. (1988) Fixtureless assembly manufacturing. *Manufacturing Eng*, 100:49-54.
4. Ngyuen, W. & Mills, J.K. (1996). Multi-robot control for flexible fixtureless assembly of flexible sheet metal autobody parts. *IEEE International Conference on Robotics and Automation*, San Francisco, California, 3, 2340–2345.
5. Langley, C.S. & D'Eleuterio, G.M.T. (2003). ART FCMAC: a memory efficient neural network for robotic pose estimation. *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Kobe, Japan, 1, 418–423.
6. Aguado, A.S., Montiel, E., & Nixon, M.S. (2002). Invariant characterization of the Hough Transform for pose estimation of arbitrary shapes. *Pattern Recognition*, 35(5), 1083–1097.
7. Chin-Hsiung, W., Shi-Jinn, H., & Pei-Zong, L. (2001). A new computation of shape moments via quadtree decomposition. *Pattern Recognition*, 34(7), 1319-1330.
8. Best, P.J. & McKay, N.D. (1992). A Method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 239–256.
9. Bribiesca, E. (1999). A new Chain Code. *Pattern Recognition*, 32(2), 235–251.
10. Gonzalez-Galvan, E.J., Korde, U.A., Chen, W., & Skaar, S.B. (1997). Application of a Precision-Enhancing Measure in 3D Rigid-Body Positioning using Camera-Space Manipulation. *International Journal of Robotics Research*, 16(2), 240–257.
11. Bone, G.M. & Capson, D. (2003). Vision-guided fixtureless assembly of automotive components. *Robotics and Computer-Integrated Manufacturing* 19(1-2), 79–87.
12. Chen, K. (1990). Efficient parallel algorithms for computation of two-dimensional image moments. *Pattern Recognition* 23(1-2), 109–119.
13. Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J.H., & Rosen, D.B. (1992). Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps. *IEEE Transactions on Neural Networks*, 3(5), 698–713.
14. Carpenter, G.A. & Grossberg, S. (2002). Adaptive Resonance Theory. *The Handbook of Brain theory and neural networks*, 2nd ed., (87-89), MIT Press.



Ismael López-Juárez obtained a BEng from UNAM in 1991. He obtained an MSc in at University of Manchester in 1996 and a PhD in Intelligent Robotics at The Nottingham Trent University, both in the U.K. His main research interests are in the areas of Neural Networks, Industrial Robots and Machine Vision. Currently, he is a Principal researcher at CINESTAV and member of the National Research System in Mexico (SNI, level 2).



Reyes Rios-Cabrera obtained a BEng from the Autonomous University of Queretaro in 2004. He obtained an MSc in Mechatronics at Aachen University of Applied Sciences, Germany in 2007. He worked as Research Assitant at Cinvestav during 2007-2008 and currently he is a student at Leuven University

pursuing his PhD in machine vision applications using multiple view scenes.



Mario Peña-Cabrera

graduated with a BEng at UNAM. He holds an MEng in Electronics from the same University and an MEng from The McMaster University, Canada. He obtained his DSc degree from PICyT, SEP-CONACYT, Mexico. His main areas of interest are in robot vision, automation and digital control, areas in which he has been working for over 30 years. Currently, He is a researcher at the Computing Engineering Science and Automation Department at IIMAS-UNAM.



Gerardo Maximiliano Méndez

received the B.Sc. degree in Electronics from the ITESM in 1985 an MBA from the UANL in 1996, and the Doctoral degree of Mechatronics Engineering from CIDESI Conacyt, Mexico, in 2005. He is currently Professor at the Electrical and Electronics Engineering Department at the Instituto Tecnológico de Nuevo Leon and his research interests are in the fields of fuzzy controllers and uncertain industrial process, modeling and control.



Roman Osorio

obtained a degree of Mechanical and Electrical Engineering, Electronics area in 1985, at the National Autonomous University of Mexico, currently conducts research in the area of mobile robotics, vision and automation in the Computing Engineering Science and Automation Department at IIMAS-UNAM.

Article received on 11/16/2010; accepted on 24/09/2012.