01 Jan 1992

Moving Object Recognition and Guidance of Robots Using Neural Networks

Abhijit Neogy

S. N. Balakrishnan
Missouri University of Science and Technology, bala@mst.edu

Cihan H. Dagli
Missouri University of Science and Technology, dagli@mst.edu

MOVING OBJECT RECOGNITION AND GUIDANCE OF ROBOTS USING NEURAL NETWORKS

Abhijit Neogy, S.N.Balakrishnan, and Cihan H. Dagli
University of Missouri-Rolla
Rolla, MO-65401

ABSTRACT

The design of a robust guidance system of a robot is discussed in this study. The two major tasks for this guidance system are the on-line recognition of a moving object invariant to rotation and translation and tracking the moving object using a neural-network driven vision system. This system includes computer software ported to the IBM PC and interfaces with an IBM 7535 robot. The operation of this guidance system involves recognition of a moving object and the ability to track it till the robot end effector is in close proximity of the object.

Simulations and testing of this system show good promise.

I. INTRODUCTION

Addition of a vision system to a robot enhances its flexibility tremendously to operate in different environments. Without vision capabilities, the orientation and the location of an object to be picked up by a robot are fixed. In addition , for each different object the robot has to be reprogrammed. With a neuro-vision system based robot proposed here, these restrictions can be overcome. In this system, the robot is capable of seeing and recognizing a part, and coordinating itself to intercept it.

A number of papers in the current literature address the problem of object recognition[1,2,3,4,5]. However, only a few of them relate to recognition of objects invariant to rotation, distortion, and translation. Mask matching techniques[1,2] can be applied to recognition of objects even though this technique is computation intensive. For a rotated object, the same technique can be used by having different masks for different orientations of the same object. This process leads to a large number of masks for a single object. Hence, we need to find efficient stand-alone algorithms robust to rotation and to translation for real time use in the vision systems.

The problem of invariant recognition can also be solved by parallel distributed processing techniques commonly known as neural networks[6,7]. Rotation invariant recognition was attempted by Chihwen, Chwan [7]. In their study, a multilayer neural network for two dimension pattern recognition with rotation, scaling, and distortion invariance is described. The rotation layer rotates the original image in polar coordinates till it matches the original template. This method requires an off-line training and there is no capability for on-line learning. If a new pattern is presented during a recall, then the network fails to recognize it, or classify it into a new category. In addition, this method consists of shifting the image itself. This means each pixel has to be displaced in the polar coordinates which makes this approach computation intensive. In our study we do not need such complex methodology.

Rumelhart[8] explains the architecture of some of these parallel processing techniques which can be used to map one vector space to another vector space. The most commonly used parallel distributed processing paradigm for pattern recognition is backpropagation. However, it is difficult to evaluate the training parameters like learning rate, number of layers, number of hidden nodes, number of iterations etc. Also, backpropagation needs supervised training. Hence, for dynamic pattern recognition backpropagation is not an ideal choice.

For recognition of objects in motion, we need efficient classifiers which can operate on-line and are rotation-translation invariant. For this purpose, we use adaptive resonance theory (ART) networks [9,10] in conjunction with image processing techniques. The image processing techniques are used to form an invariant vector which is input to ART for efficient on-line recognition.

In this study, we investigate the motion of objects in arbitrary dynamic conditions and the use of robot to track and intercept them. For tracking of the moving object we need camera-robot coordination. This task requires a non-linear transformation of the camera coordinates in pixel to the robot coordinates in millimeters. Image processing techniques in tandem with backpropagation are used to achieve this mapping.

The rest of the paper is organized as follows: we present the architecture of the proposed vision based guidance system in section II. The detailed descriptions regarding design and implementation of each of the sub-systems are presented in section III. Experimental validation of this neuro-vision system is described in section IV. Conclusions are presented in section V.

II. ARCHITECTURE OF THE VISION BASED GUIDANCE SYSTEM

The architecture of the proposed guidance system of the robot consists of two major modules. The first module consist of sub-systems related to the recognition of the object, the second module is related to the tracking of the object in the robot work space.

The recognition module includes a vision system to capture the images of a moving object. Subsequently image processing algorithms are used for detection of edges. Encoding algorithm are used to extract significant features from the output of image processing algorithms to be used as input to the neural network for object recognition.

The task of the tracking module involves determination of

 CH3158-3/92/0000-0883 $1.00

the dynamics and interception of the object. This module consist of a calibration system to map the camera coordinates to the robot coordinates and a prediction sub system to predict the dynamics of the object. The block diagram of the architecture is presented in figure 1.

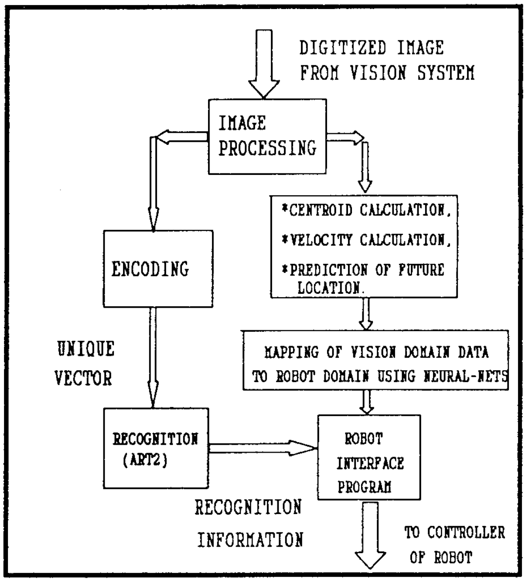


Figure 1 Architecture of the robot guidance system

III. DESIGN AND IMPLEMENTATION OF THE NEURO BASED VISION GUIDANCE SYSTEM

1. Recognition Module

The vision system

This system consists of a camera mounted on the robot end-effector and a vision card. The images from field of view of the camera is displayed on the monitor attached to the camera. The image captured by the camera is stored as an ASCII file in the RAM of the vision card. The vision card is mounted on the controlling personal computer. This vision card has a RAM of 640*480 bytes.

The image processing algorithms

Image processing algorithms used in this architecture perform the task of noise filtering, edge detection and thresholding.

Noise filtering

The digital file representing the image captured by the camera and stored in the vision card can be quite noisy. In order to smooth out noise, a Gaussian filter is used. The Gaussan filter is a low pass filter. Useful information is mostly concentrated in the region of lower frequency and noise characteristic usually appear in the higher frequency region. Hence, filtering of high frequency signals results in elimination of noise. Though some useful signals get filtered out in the process, this loss of information has very little impact on our study. The Gaussian mask is implemented using a Gaussian function[2]. This mask is passed over the ASCII image file which is stored in the vision card.

Edge detection

The aim of this task is to detect the boundary of the object in the work place. After the noise from the digitized file is removed, a Laplacian filter is passed over the data file. Lapalacian filter is a high pass filter. An edge in a digitized image file is a large transition from a low pixel value to a high pixel value or vice-versa. Hence, a large change in adjacent pixel values can be viewed as a high frequency. On passing the digitized file through a high pass Laplacian filter the edges in the image are detected [2].

The main reason for edges detection is to help locate the boundary of the object in the field of vision. Also, if all the pixels that represent the object are used we will have a large number of pixels to deal with. Through the edge detection procedure we use considerably less number of pixels for further calculations.

Through thresholding the different analog values of the pixels which represents different shades of grey are converted to bipolar values representing black and white only.

Encoding

Encoding means representing the object in the field of vision of the camera by a vector such that this vector remains approximately invariant to rotation or shifting in the field of vision. It should be noted that encoding also results in data compression because the whole data file of 65,536 bytes representing an object is now represented by a vector the dimension of which is determined by the user.

Generation of the encoding vectors is carried out in the following manner. All the pixel values are first converted to bipolar values by thresholding so that high values represent the edges of the object and low values the background. The image file now contains only the boundary of the object. The centroid of this boundary is calculated and set as the origin of a polar coordinate system. Then, the distance of the furthest edge from the centroid is found. The line joining the centroid and the furthest edge location is the reference axis of the polar coordinate system. The radial distance between the centroid and the edges of the object is calculated at pre-determined angles of rotation from the reference axis. If we decide to calculate the radial distance every 90 degrees we will be having 4 such radial distance information about the object. These four data constitute a vector representing the object are invariant to rotation and translation. The angle at which the radial distance is measured can be reduced to increase the number of radial distance information and thereby increase the accuracy of recognition. Results of the encoding method for a representative object placed in different orientations are shown in Table 1. It can be seen from Table 1 that regardless of the orientation of the object, the components of the encoding vector remains the same.

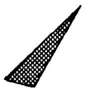
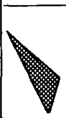
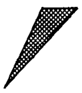
OBJECT ORIENTATION	RESULTS OF ENCODING
	CENTROID = [161 , 67] VECTOR = [59.4, 12.7, 42.4, 12.0]
	CENTROID = [120 , 61] VECTOR = [59.3, 12.7, 42.4, 11.3]
	CENTROID = [258 , 125] VECTOR = [59.4, 15.6, 39.6, 9.8]
	CENTROID = [305 , 248] VECTOR = [59.4, 10.0, 39.6, 16.27]

Table 1 Rotation invariant encoded vector.

ART2

This is an adaptive and autoassociative neural network paradigm. This paradigm can operate on-line, and highly suitable for pattern recognition and data clustering [6,7]. No previous training is required in the ART network for operation in the recall mode. When ART2 receives an input pattern, it searches for some matching between the stored patterns and this given input pattern. If a sufficiently similar pattern is found then the class representing this pattern is activated, and learning takes place. If sufficient similar match between the patterns stored and the presented pattern is not found, then a new class is created for this pattern and learning takes place for this particular output node only. The degree of coarseness of classification depends on the vigilance parameters. The vector generated by the encoding algorithm is input to ART2, for recognition of the object.

2. Tracking Module

Calibration

Calibration of the robot work space refers to mapping a location on the digitized file containing the image of the tracked object to the robot coordinates. A pixel location represented by two cartesian coordinates x,y as calculated from the digitized image file can be used to generate the corresponding robot coordinates by a non-linear transformation. This mapping enables the robot end effector to reach the same location as stored in the vision system. The process of calibration links the robot work space and the vision system.

Mathematical modelling of this non-linear transformation between the vision coordinates and the robot work space is difficult and complex. Hence we use a backpropagation network to learn this relationship through a training process with input-output pairs. A backpropagation network[8] is used for calibration. Two inputs to backpropagation network correspond to the x,y location of the object in pixel coordinates and the two outputs represent the corresponding x,y location of robot coordinates in millimeters. The single hidden layer in the network has 30 nodes. The training process requires 50,000 iterations for accurate mapping to take place, with a learning rate of 0.5.

For training the backpropagation network we have two data sets. One of the data sets contains the x,y location in pixel coordinates and the second data set contains the x,y location in robot coordinates. The input to the network are the x,y location in pixel values. The desired outputs are the corresponding x,y location of the robot coordinates. Generation of the input data is done by placing a checkered paper in the camera field of view which also overlaps with the robot work space. This picture is digitized by the vision system and stored as an ASCII file. By using a software tool, the ASCII file could be displayed and the pixel values (x,y) of the corners of the checkered boxes are noted. These coordinates, thus represent a location in the field of vision of the camera.

In order to find the corresponding robot coordinates, the IBM AML (A Machine Language) program is used and the end-effector of the robot is moved to each corner of the boxes on the checkered paper. Each time the robot coordinates are noted. In this way the data for the second data file required for training the backpropagation is generated. We now have the two input and output data sets of different coordinates each representing the same location.

The data set containing the x,y location in pixel values are the inputs to the backpropagation network and the data set containing the x,y location in robot coordinates are the desired outputs.

Prediction

For the robot end effector to be able to catch up with the moving object the delay in response time has to be taken account of in computing the dynamics of the object. This is done by moving the robot end effector to reach a predicted location based on the present dynamics. In order to be able to intercept the moving object, the centroid of the moving object at each instant is calculated. Based on the displacement from the previous location it's velocity is calculated. Assuming that the velocity is constant, the location of the object for a future time is predicted and the robot end effector is moved to that location. This sequence is repeated till the robot end effector is in close proximity to the object when grabbing sequence is initiated.

IV. EXPERIMENTAL VALIDATION OF THE NEURO--VISION SYSTEM

The guidance system developed in this study was validated by implementing with an IBM 7535 robot. Objects of different shapes are moved by using a magnet. The view area of the camera was about 13cm*9cm, which was converted into a image of 640*480 pixel. It took about 4 seconds for the system to digitize a picture, process it and move the robot manipulator to the predicted location. It was concluded that the maximum possible speed that could be imparted to the object was 13/4 = 3cm per second (approximate). This system is designed to grab the moving object only if the distance between the previous position and the present position is less than 5 pixels.

It was found that the robot was able to track the moving object as long as it did not go out from the region visible to the camera. Recognition was successful and demonstrated for objects of arbitrary shapes. Even similarly shaped objects of different sizes were correctly recognized.

V. CONCLUSION

A neural network robot guidance system has been developed. This system is demonstrated to be capable of tracking and two dimension shapes recognition of an object in motion, invariant to rotation and translation.

Two neural network paradigms, Backpropagation and ART2 have been implemented using object oriented programming techniques. Backpropagation was successful in achieving a non-linear mapping, between the vision to robot coordinates. The ART2 neural network paradigm was able to achieve very fine resolutions in classification and could differentiate between similar shapes of different sizes.

The limitations in speed of performance of such system are due to the memory limitations and the clock frequency of the controlling computer.

REFERENCES

1. Tzay Y. Young, King-Sun Fu,"Handbook of Pattern Recognition and Image Processing", Academic Press Inc,1986.
2. Kenneth R. Castleman,"Digital Image processing", Prentice-Hall Inc, 1979.
3. Brian L. Pulito, Raju Damarla, Sunil Nariani, " 2-D Shift Invariant image Classification Neural Network, which overcomes Stability, Plasticity Dilemma", Vol 2, International Joint Conference on Neural Network, San Deigo, June 17-21,1990.
4. M. Marefat and R. L. Kashyap, "Image Interpretation and Object Recognition in Manufacturing", IEEE International Conferance on Systems, Man, and Cybernatics, Los Angeles, CA, November 4-7, 1990.
5. Jon M. Barrilleaux, " A biblogically motivated algorithm for image interpretation based on multi-pass, multi-resolution technique",Vol 2, International Joint Conference on Neural Network, San Deigo, June 17-21,1990.
6. Wei li, Nasser M. Nasrabadi,"Invariant Object Recognition Based on Neural Network of Cascade REC Nets", Vol 2, International Joint Conference on Neural Networks, San Deigo, June 17-21,1990.
7. Chihwen Li, Chwan Hwa Wu,"A Rotation, Scaling, and Distortion Invariant, Pattern Recognition Neural-Networks", Workshop for neural networks: Academic/Industrial/NASA/Defense, Auburn, Feb 11-13,1991.
8. D.E Rumelhart, J.H McClelland and the PDP Research Group,"Parallel Distributed Processing", Vol 1,2. Cambridge, MA: MIT Press, 1986.
9. G. A. Carpenter and S. Grossberg,"ART2: self-oganization of stable category recognition codes for analog input patterns.", Vol 26, Applied Optics,December 1987.
10. G.A Carpenter and S. Grossberg, " The ART of Adaptive pattern recognition by a self- organizing Neural Network", COMPUTER, March 1988.
11. L.D Jackel, B.Boser, J. S. Denker, H. P. Graf, Y. Le Cun, I.Guyon, D. Henderson, R.E. Howard, W.Hubbard, S. A. Solla," Hardware Requirements for Neural-Net Optical Character Recognition", Vol 2, International Joint Conference on Neural-Networks,San Deigo,June 17-21,1990.
12. J.F. Kolen and A. K. Goel ,"Learning in Parallel Distributed Processing Network: Computational Complexity and Information Content",Vol 21, IEEE Transactions on Systems, Man, and Cybernatics, March/April 1991.
13. Alan A.Yamamura, Athanasios Sideris, Chuanyi Ji, and Demetri Psaltis, "Neural-Network Control of a Two-Link Manipulator," Proceedings of the 29th Conference on Decision and Control, Honolulu, Hawai, December 1990.
14. Mo. Jamshidi, Bill Horne, Nader Vadiee, " A Neural Network Based Controller for a Two-Link Robot", Proceedings of the 29th Conference on Decision and Control, Honolulu, Hawai, December 1990.
15. Sabri A. Mahmoud, Roger J. Green, "Identification and Velocity Computation of Multiple Moving Objects in Images",Vol 26, NO.4, IEEE Transctions on Aerospace and Electronics Systems, July 1990.
16. Alan A.Yamamura, Athanasios Sideris, Chuanyi Ji, and Demetri Psaltis, "Neural-Network Control of a Two-Link Manipulator," Proceedings of the 29th Conference on Decision and Control, Honolulu,Hawai,December 1990.