International Journal of Systems Science, 2001, volume 32, number 9, pages 1155–1159

Camera calibration and three-dimensional world reconstruction of stereo-vision using neural networks

Qurban Memon† and Sohaib Khan‡

Stereo-pair images obtained from two cameras can be used to compute three-dimensional (3D) world coordinates of a point using triangulation. However, to apply this method, camera calibration parameters for each camera need to be experimentally obtained. Camera calibration is a rigorous experimental procedure in which typically 12 parameters are to be evaluated for each camera. The general camera model is often such that the system becomes nonlinear and requires good initial estimates to converge to a solution. We propose that, for stereo vision applications in which real-world coordinates are to be evaluated, artificial neural networks be used to train the system such that the need for camera calibration is eliminated. The training set for our neural network consists of a variety of stereo-pair images and corresponding 3D world coordinates. We present the results obtained on our prototype mobile robot that employs two cameras as its sole sensors and navigates through simple regular obstacles in a high-contrast environment. We observe that the percentage errors obtained from our set-up are comparable with those obtained through standard camera calibration techniques and that the system is accurate enough for most machine-vision applications.

1. Introduction

Camera calibration is considered as an important issue in computer vision. Accurate calibration of cameras is especially crucial for applications that involve quantitative measurements, depth from stereoscopy or motion from images. The problem of camera calibration is to compute the camera extrinsic and intrinsic parameters. The extrinsic parameters of a camera indicate the position and the orientation of the camera with respect to the coordinate system, and the intrinsic parameters characterize the inherent properties of the camera optics, including the focal length, the image centre, the image scaling factor and the lens distortion coefficients. The number of parameters to be evaluated depends on the camera model being utilized. Typically, 12 parameters are found for each camera, expressed as

$$C_{\mathrm{h}} = A W_{\mathrm{h}}, \tag{1}$$

where

$$\begin{bmatrix} C_{\mathrm{h1}} \\ C_{\mathrm{h2}} \\ C_{\mathrm{h3}} \\ C_{\mathrm{h4}} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{2}$$

The $C_{\mathrm{h1}}, C_{\mathrm{h2}}, C_{\mathrm{h3}}, C_{\mathrm{h4}}$ are known as camera coordinates, $W_{\mathrm{h}} = [X\ Y\ Z\ 1]^{\mathrm{T}}$ are known as world homogeneous coordinates and A denotes the unknown (12 parameters) camera matrix. The problem of finding these parameters is, in general, a nonlinear problem (owing to lens distortion) and requires good initial estimates and an iterative solution.

The techniques found in the literature for camera calibration can be broadly divided into three types: linear methods, nonlinear methods and two-step techniques. Linear methods assume a simple pinhole camera model and incorporate no distortion effects. The algorithm is non-iterative and therefore very fast (Abdel-Aziz and Karara 1971, Wong 1975, Ganapathy 1984, Frugeras and Toscani 1986). The limitation in this case, however, is that camera distortion cannot be incorporated and therefore lens distortion effects cannot be

Received 21 July 1998. Revised 14 November 2000. Accepted 16 August 2000.

† Hamdard Institute of Information Technology, Hamdard University. Madinat-al-Hikmah Avenue, Karachi 74600, Pakistan. Fax. 92-21-4543878, e-mail: qurban@hamdard.net.pk.

‡ School of Computer Science, University of Central Florida, Orlando, FL 32816, USA.

corrected. The problem of lens distortion is significant in most off-the-shelf charge-coupled device cameras. In non-linear techniques, first the relationship between parameters is established and then an iterative solution is found by minimizing some error term. Many classical calibration techniques fall in this category (Brown 1966, Haralick and Shapiro 1993, Nomura *et al*. 1992). Direct linear transformation introduced by Abdel-Aziz and Karara (1971) has also been extended to incorporate distortion parameters. The advantage of such techniques is that the camera model can be very general to accommodate different types of camera. However, for this type of an iterative solution, a good initial guess is essential, otherwise the iterations may not converge to a solution. Two-step techniques involve a direct solution of some camera parameters and an iterative solution for the other parameters. Iterative solution is also used to reduce the errors in the direct solution. This is the most common and current approach to the problem (Tsai 1987, Lenz and Tsai 1988, Weng 1992).

Computing world coordinates from stereo images requires first matching the images obtained from two different cameras to determine disparities (difference in positions of corresponding features) and then transforming these into world distances. The problem has been called the object pose estimation problem in computer vision literature (Haralick and Shapiro 1993) or simply the stereo reconstruction problem. The process of matching is essential for finding world coordinates from a stereo image. The matched points are used to find world coordinates using triangulation (Gonzalez and Woods 1992). In this process, all the camera calibration parameters appear as constants in the equation. Hence, camera calibration is essential to compute world coordinates from stereo-images.

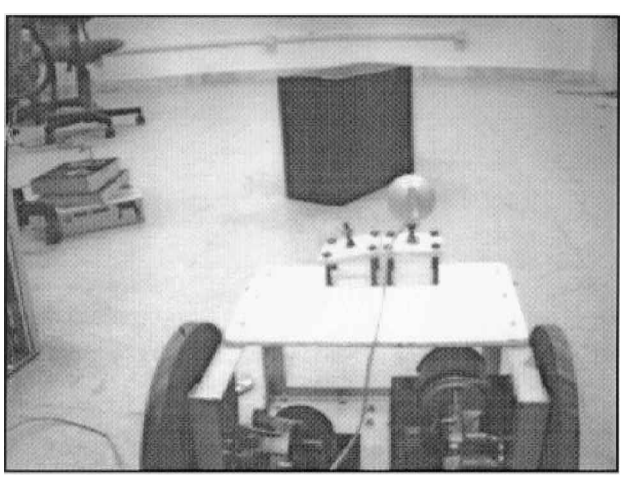

Figure 1. Mobile robot. The objects in front of the camera are obstacles placed for the purpose of experiment.

In the next section, § 2, we present a simple and unified approach to camera calibration and stereo reconstruction using neural networks. In our approach, instead of calibrating both cameras and then using the triangulation procedure, we directly train a neural network to compute world coordinates from matched pairs of image points. The advantage that we obtain is that the approach is not dependent on the camera model and will work for any type of camera. Figure 1 shows the mobile robot used in our experiment with one camera mounted on it. In § 3, we discuss the results obtained by our approach, when tested on our prototype mobile robot system. In § 4 we present conclusions.

2. Artificial neural network model for three-dimensional world reconstruction

Artificial neural networks (ANNs) are being applied in many scientific disciplines to solve a variety of problems in pattern recognition, prediction, optimization associative memory and control. None of the conventional approaches to these problems is flexible enough to perform well outside their domain. ANNs provide exciting alternatives and many applications could benefit from them (Jain, A., *et al*. 1996).

In our problem, we propose a multilayer ANN model because camera calibration problem is a nonlinear problem and cannot be solved with a single layer network (Fausett 1994). The best architecture and algorithm for the problem can only be evaluated by experimentation and there are no fixed rules to determine the ideal network model for a problem. However, variations in architecture and algorithm effect only the convergence time of the solution.

We have used the network model in figure 2 for our simulations. It falls into the category of the feedforward class. Each output in a layer is connected to each input in the next layer. In this case, the output layer has simple linear neurons, while all the neurons in the two hidden layers have the same transfer function, with a sigmoidal nonlinearity. Generally, the nonlinear, continuously differentiable, real-valued and bounded function for three inputs and their corresponding weights is shown in figure 3, where the parameter a differentiates from hard limiting function. Also, because there is no feedback between layers, the effect of the feedforward neural net topology is to produce a nonlinear mapping between the input nodes and the output nodes. The model that we used consists of four input neurons, eight hidden neurons and three output neurons. The input neurons correspond to the image coordinates of matched points found on the stereo images (x_1, y_1) and (x_2, y_2). These points are generated by the same world point on both images and form the input of the neural network. The output neurons correspond to the actual world coordi-

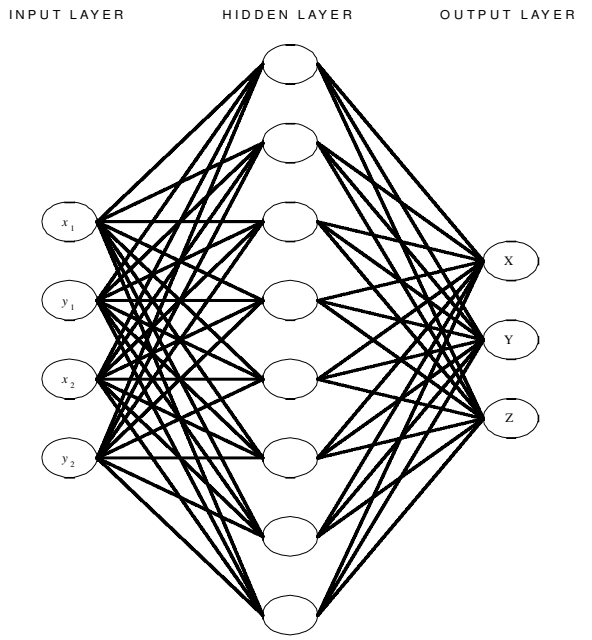

Figure 2. ANN model used for the problem.

nates of the point (X, Y, Z) which are mapped as (x_1, y_1) and (x_2, y_2) on the two images. We train the network on a range of inputs and outputs, such that the network could, after training, give the world coordinates for any matched pair of points. The implementation details and the results are given in the next section.

The approach requires training of the network for a set of matched image points whose corresponding world point is known. For this purpose, we use an object similar to that used by (Nomura *et al.* 1992) consisting of a grid of points placed at fixed intervals. This chart (shown in figure 4) is placed in front of the two cameras at known distances from an arbitrary world origin. It should be noted that the choice of the world origin in this approach is arbitrary and the cameras need not be fixed at some precise location relative to the world origin. We capture stereo images of the chart at various distances from the world origin, noting the value of the

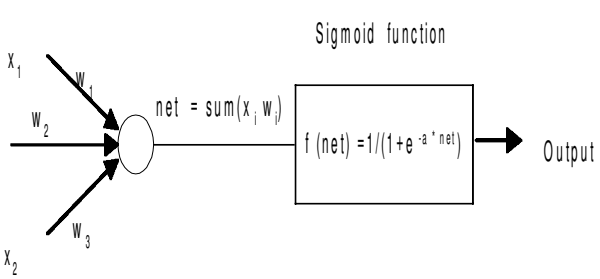

Figure 3. Neuron model with sigmoid function.

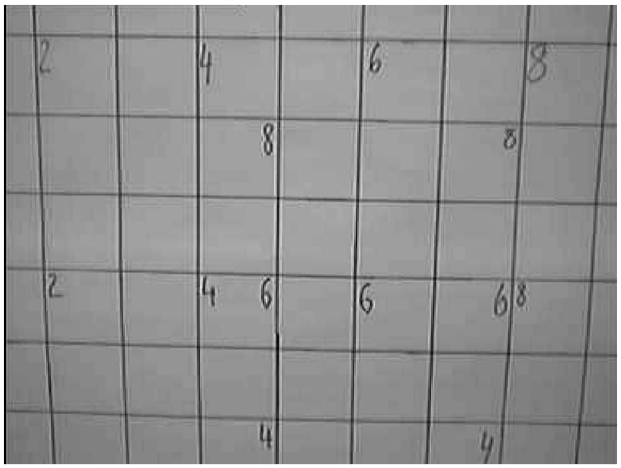

Figure 4. Calibration chart at 50 cm.

world coordinates of the chart at each instance. The set of matched points and the world coordinates thus obtained form the training set of our ANN. Once the network is trained, we present it with arbitrary matched points and it directly gives us the world coordinates corresponding to the matched pair.

It should be noted that this approach is different from conventional camera calibration techniques in the sense that no extrinsic or intrinsic camera parameters are found for any of the cameras. Instead, the system is trained such that it learns to directly find the world coordinates of objects. The experimental procedure required is almost the same as that of conventional approaches to the problem. However, the approach is essentially very simple and yields comparable results.

The advantage of this approach lies mainly in its simplicity and generality. The technique will work for any type of camera and accurate camera modelling is not an issue. The cameras need not be fixed at any precise location with respect to the world origin, nor do their axes have to be aligned. The precise positioning of the chart with respect to the camera is also not required, as is the case in some approaches to the problem (see, for example, Nomura *et al.* (1992)). The calibration chart only needs to be at known positions with respect to a world origin.

It should be noted that this approach of camera calibration is only valid for stereo vision systems and is not applicable to monocular cameras. The approach is particularly suited to autonomous mobile robots that employ stereo vision. It is novel in the sense that it is based on training rather than computing explicit values of camera parameters. However, the training set presented to the ANN must be a good enough representative of the range of possible scenarios that the system might encounter during operation.

3. Experimental details and results

We took two cameras mounted on our prototype mobile robot. We kept the distance between the two cameras at approximately 7 cm and did not align their optical axes precisely. Next we made a calibration chart consisting of a grid of lines 5 cm apart (as shown in figure 4). This chart was placed in front of the cameras at various distances from the world origin and its image captured from both cameras, without moving the cameras. The chart was placed at distances that were in the range of interest of our robot. We defined the range of interest of the robot to be within 50–140 cm in front of the robot and captured images in this range at increments of 15 cm. We felt that our robot should be able to gauge correctly the distance of obstacles that are present in this range.

After capturing the images of the calibration chart, we matched these images to obtain stereo pair points. For each stereo pair, we also knew the actual world distance, since we had placed the chart at measured distances with respect to a world origin. At no time in our experiment did we have to measure the exact distance of the cameras with respect to the world origin, as would have been necessary in some calibration approaches found in literature.

A set of 400 stereo pairs and their actual three-dimensional (3D) world distances formed our training set. We trained our neural network on this set of 400 stereo pairs. The training was done by presenting the stereo-pair points to the input of the network and presenting the 3D world coordinates at the output.

The training was done using the back-propagation algorithm (briefly described in table 1), with Nguyen–Widrow initialization of weights and adaptive learning model. We used a log-sigmoid activation function for both inputs and weights. All inputs were normalized between 0 and 1 before presenting them to the network. The target outputs were also normalized between 0 and 1. Such normalization is necessary to obtain quicker learning. We experimented with various different network architectures but observed very little change in error by using alternative architectures.

To check the accuracy of the trained network, we presented the network with stereo-pair points that were not included in the training set but were from within our range of interest of distance. We had a set of such points whose corresponding 3D world coordinates were already known to us. We computed the average error that we obtained from these points. This error presented the true learning of the network, since we had not included these points in the training set. Mathematically, this mean square error can be written as

$$e_{\text{ms}} = \text{mean } [(x - \hat{x})^2], \tag{3}$$

where e_{ms}, x and $\hat{x}$ denote the mean square error of the network, the world point vector that is actually measured and the corresponding world point vector given by the network respectively.

The results of the mean percentage error observed during training and computed as a percentage are shown in figure 5. From figure 5 it can be seen that, as the training epochs are increased, the error in the computations made by the network is decreased. The error became less than 5% after 40 000 epochs of training. After 100 000 epochs of training, the mean error in computing 3D coordinates of a point became 4.33%.

It must be appreciated that this error contains not only the error that is contributed by the network but also the quantization errors of the camera. Since we did not use any subpixel measuring technique to find the image coordinates of a point, we should have a significant contribution of quantization error.

Since the sigmoid activation function has also been applied on outputs and the training data have been normalized, we wanted to verify the ability to extend linearly the output range. Once our network was trained in our range of interest, we also presented it with points that were outside our range of interest. We had originally trained the network for points within 140 cm of the

Table 1. Back-propagation algorithm steps

(1)	Initialize weights
(2)	Present input and desired output
(3)	Calculate actual outputs
(4)	Adapt weights using recursive algorithm starting at the output nodes and working back to the first hidden layer to adjust weights

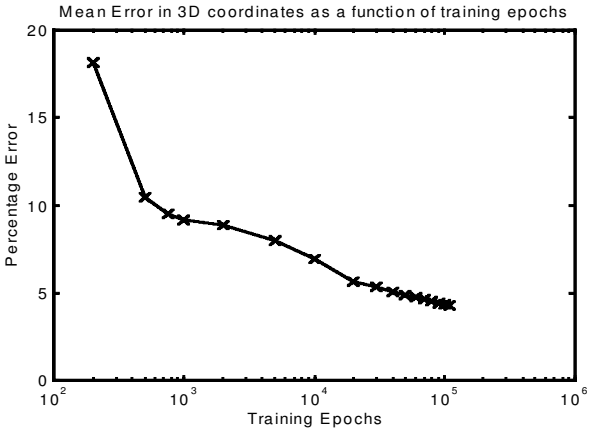

Figure 5. Mean percentage error in computing 3D coordinates as a function of the number of epochs.

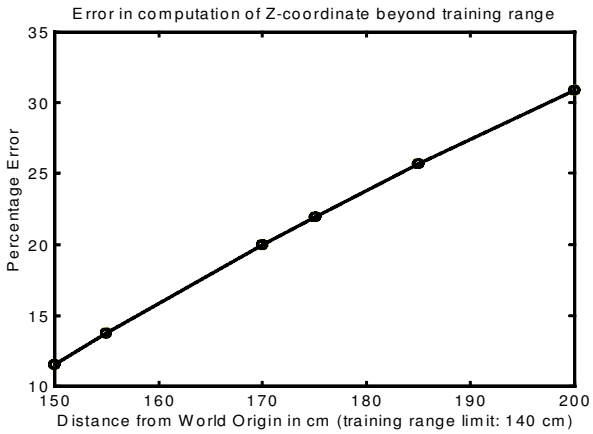

Figure 6. Percentage error in the computation of the Z coordinate beyond the training range (results taken after training the network for 50 000 epochs).

world origin. Now we presented the network with points whose distance from the world origin ranged from 150 to 200 cm and again computed the percentage error using equation (3), but this time these results were accumulated as a function of distance beyond training set. These results are shown in figure 6. We observed a linear increase in percentage error as the object moved further away from our training limit. Now, the inverse perspective transform equation that is used to compute the world distance in normal circumstances (i.e. not using our approach, but instead using conventional methods) implies that the error should increase as a square of the distance. That is because there is $1/Z$ term in that equation, and the magnitude of error comes out to be a $1/Z^2$ term. It is also what one would expect, since two cameras are used. There is no doubt that the error would also increase in a quadratic fashion if we had been using a simple interpolation scheme, but that is not the case in our approach. Neural networks are supposed to behave well in their training region and close to the edges of their training region. So by virtue of using a neural network, our error is increasing in a linear fashion, which is an actual improvement over the interpolation approach. One important point here is that the interpolation approach will obviously give exact results at points that are part of the training set. A trained neural network might not give an exact reconstruction for these points, that is it has an error even for the points that are actually part of the training set and, as we have seen, better comparative performance outside the training set. It should be noted that we placed minimal constraints on the type of camera required, the resolution and quality of images and the accuracy of measurements.

4. Conclusion

In this paper, we have presented a unified approach to camera calibration and 3D world reconstruction for stereo-vision. We used an ANN to train the system such that, when the system is presented with a matched pair of points, it automatically computes the world coordinates of the corresponding object point. The approach differs from conventional approaches to the problem, which appear in computer vision literature in the sense that the cameras are never actually calibrated, and the network is so trained as to compute the correct world coordinates of two matched points. The approach is simple in concept, independent of the camera model used and the quality of image obtained and yields very good results when applied to a prototype autonomous mobile robot using stereo-vision.

References

Abdel-Aziz, Y., and Karara, H., 1971, Direct linear transformation into object space coordinates in close-range photogrammetry. *Proceedings of the Symposium on Close-Range Photogrammetry*, pp. 1–18.

Brown, D., 1966, Decentering distortion of lenses. *Photogrammetric Engineering Remote Sensing*, 444–462.

Fausett, L., 1994, *Fundamentals of Neural Networks: Architectures, Algorithms and Applications*, (Englewood Cliffs, New Jersey: Prentice-Hall), pp. 289–330.

Frugeras, O., and Toscani, G., 1986, Calibration problem for stereo, *Proceedings of the International Conference on Computer Vision Pattern Recognition*, pp. 15–20.

Ganapathy, S., 1984, Decomposition of transformation matrices for robot vision, *Proceedings of the IEEE International Conference on Robotics and Automation*, (New York: IEEE), pp. 130–139.

Gonzales, R., and Woods, R., 1992, *Digital Image Processing*, (Reading, Massachusetts: Addison-Wesley), pp. 56–71.

Haralick, R., and Shapiro, L., 1993, *Computer and Robot Vision*, Vol. 2 (Reading, Massachusetts: Addison-Wesley), pp. 125–178.

Jain, A., Mao, J., and Mohiuddin, K., 1996, Artificial neural networks: a tutorial. *IEEE Computer Magazine*, 31–44.

Lenz, R., and Tsai, R., 1988, Techniques for calibration of the scale factor and image center for high accuracy 3D machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**, 713–720.

Nomura, Y., Sagara, M., Naruse, H., and Ide., A., 1992, Simple calibration algorithm for high-distortion lens camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**, 1095–1099.

Tsai, R., 1987, A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal on Robotics and Automation*, 323–344.

Weng, J., Cohen, P., and Herniou, M., 1992, Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* **14**, 965–980.

Wong, K., 1975, Mathematical formulation and digital analysis in close range photogrammetry. *Photogrammetric Engineering Remote Sensing*, **41**, 1355–1373.