A New Neural Network Structure for Camera Calibration

YONGTAE DO
School of Computer and Communication Engineering
Daegu University
Jinryang, Kyungsan-City, Kyungpook, 712-714
Korea

Abstract: - In this paper, a new design of neural network is proposed for camera calibration. Unlike most existing methods, where camera parameters are determined so that the projection from 3D points to their corresponding image points is as accurate as possible, the network of multilayer perceptrons designed here learns the transformation from image points to their corresponding rays of sight. Since this is a one-to-one mapping, whereas the projection is a many-to-one mapping, the transformation and its applications are quite straightforward. Being based on a geometric model, a network design process also does not require a tedious step of determining the numbers of hidden layers and nodes for efficient learning with given data.

Key-Words: - Camera calibration, neural network, multilayer perceptrons, projection, back-projection

1. Introduction

In general, vision is the most important and useful sense for both humans and machines. For successful visual sensing, an accurate mapping between the space viewed and corresponding image captured is important. Humans and animals learn or possess this mapping capability by nature. For machines, however, the intrinsic and extrinsic parameters of cameras should be computed implicitly or explicitly to determine the mapping before they are used for visual perception. This process called camera calibration is thus a key step for further processing in most 3D machine vision applications.

Although the problems of stereo and motion have been with the major research interests in the field of 3D vision, it was pointed out that camera calibration is even more important practically than these noble problems by two reasons [1]:

- Information obtainable by calibration is a prerequisite for all stereo algorithms
- Calibration is basically the same as estimating the motion of a camera

A considerable number of camera calibration techniques have been proposed and they can be classified by different criteria. For example, a technique may be implicit or explicit [2], linear or nonlinear [3], and analytic or iterative [4]. Since every technique has its own advantages and disadvantages, no one can be the absolute best in different conditions and applications. Comprehensive study on existing camera calibration techniques can be found in [3-5].

Whilst camera calibration was arisen as an important procedure for 3D vision tasks and attracted the attention of many researchers, a great interest was given also to artificial neural networks (ANNs) as they were successfully applied for various problems. Therefore, as a natural consequence, some researchers tried to employ an ANN for the problem of camera calibration.

There are mainly three different approaches in neural camera calibration. First, an ANN can be used jointly with an existing non-neural calibration technique especially to compensate for some shortcomings of the non-neural one [6-8]. Since a neural learning is basically an implicit modeling, using it with an explicit calibration method has practical advantages. This is probably the most popular way of using ANNs when they are employed in camera calibration but the role of ANN is minor. The same approach can be also used for stereoscopic back-projection problem [9].

The second approach is using only an ANN for solving the problem. This is simple in concept. However, the learning is too slow practically and hard to arrive at small error in reasonable time. This may be because the projection transformation from

3D points to image points is a many-to-one mapping; different 3D points may correspond to the same image point. Thus, stereoscopic back-projection, where a 3D point is uniquely determined from two matched image points, is more appropriate application in this approach [10].

The third approach is using a network designed to be capable of explicit calibration. The ANN presented by Ahmed and his colleagues might be the first of its kind [11,12]. Since the network was designed based on a physical model, the weights of the network synapses were related directly to the position, orientation and optical parameters of a camera calibrated. Therefore, no need for searching a good network structure is required unlike other techniques employing ANNs.

This paper describes a new design of ANN for camera calibration. Like Ahmed's it is designed based on a physical camera model and the network can tell camera parameters explicitly. Therefore, all advantages of Ahmed's approach can be found here also. However, unlike almost all existing techniques including Ahmed's, where calibration is done by optimizing the mapping from 3D points to their corresponding 2D image points, the technique proposed in this paper learns the mapping from 2D or 3D points to their rays of sight. Since this is a one-to-one mapping uniquely determinable when a point is given, the projection and back-projection become straightforward and easy to be done.

2. Neural Network Design

2.1 Camera model

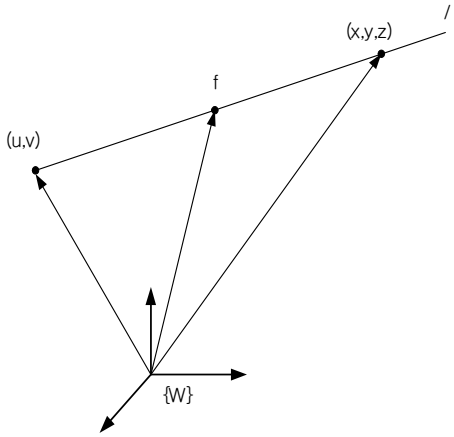

Fig.1. Pinhole camera model

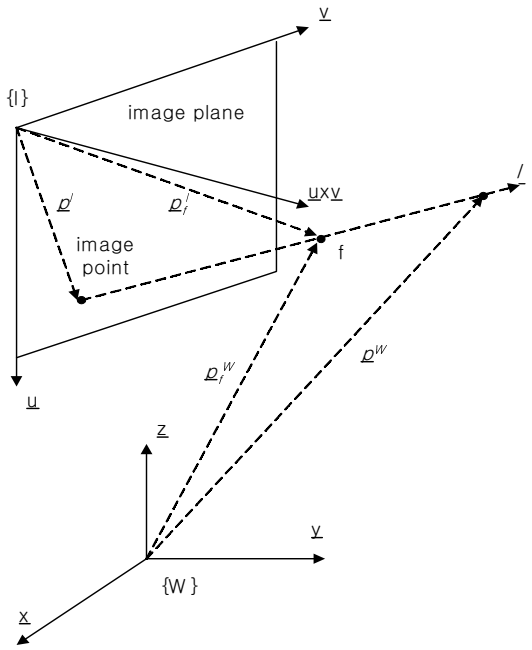

Fig.2. A ray determined from an image point

Assuming pinhole camera model a 3D point at (x,y,z), its corresponding image point at (u,v), and the pinhole at focal distance *f* all are on the same ray *l* in the world coordinate system {W} as shown in Fig.1. Therefore, if we know two of the three points, the ray can be uniquely determined. Especially, if the focal point is known, we can find the ray from either an image point or a 3D point.

Assuming a 3D frame {I} attached to the image plane as shown in Fig.2, an arbitrary image point is represented as $\underline{p}^I = (u, v, 0)^T$ in {I}. From the image point, a ray of sight can be determined as it passes the focal point $\underline{p}_f^I = (u_0, v_0, f)^T$ in {I}, which can also be represented as $\underline{p}_f^W = (p_{fx}, p_{fy}, p_{fz})^T$ in {W}. The aiming vector of the ray is defined then as

$$\underline{a}^I = \underline{p}_f^I - \underline{p}^I \quad (1)$$

In {W}, this is

$$\underline{a}^W = \underline{R}_{WI}\,\underline{a}^I \quad (2)$$

where $\underline{R}_{WI} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$ is a rotation matrix from {I} to {W}. As the ray pass a 3D point $\underline{p}^W = (x, y, z)^T$ in {W}, that is projected onto the

image point, the equation defining a ray of sight from the *t*'th image point $\underline{p}_t^I$ becomes

$$\underline{p}_t^W = \underline{p}_f^W + s_t \underline{R}_{WI}(\underline{p}_f^I - \underline{p}_t^I) \quad (3)$$

where s_t is a scale factor representing the ratio between lengths of aiming vector and a vector to $\underline{p}_t^W$ from the pin-hole. This is not a constant but it needs not to be learned exactly in calibration process as we try to find a ray rather than a point from a given image point. The only condition we will impose on s during the network learning is that it is a positive constant minimizing the distance between a 3D point and the ray from its image point.

2.2 Neural network structure

Fig.3 shows the neural network implementation of Eq.(3) derived. The outputs of the first and the second hidden layers are the aiming vectors in {I} and {W} respectively. The output of the total network is the coordinates of a 3D point, that is on the ray of sight from the image point given.

The network can be trained by the error back-propagation algorithm so that the following error function is minimized for N data given

$$E_t = \frac{1}{2}\sum_{n=1}^{3}(o_{nt} - p_{nt}^W)^2, \qquad t = 1,\cdots,N \quad (4)$$

where o_{nt}, $n = 1,\cdots,3$, are the outputs of the network for *t*'th data. Assuming all linear activation functions, parameters are modified iteratively to reduce the error function by

$$\frac{\partial E_t}{\partial p_{fn}^W} = o_{nt} - p_{nt}^W \quad (5)$$

$$\frac{\partial E_t}{\partial r_{mk}^W} = s_t(o_{mt} - p_{mt}^W)H_{kt}^{(1)}, \quad m = 1,\cdots,3, \ \ k = 1,\cdots,3 \quad (6)$$

$$\frac{\partial E_t}{\partial w_h} = s_t\sum_{m=1}^{3}(o_{mt} - p_{mt}^W)r_{mh}, \quad h = 1,\cdots,3 \quad (7)$$

where $H_{kt}^{(1)}$ is the *k*'th output of the first hidden layer for the *t*'th datum. Note that $w_1 = u_0$, $w_2 = v_0$, $w_3 = f$. The scale factor s_t can be determined for the point like

$$s_t = (\underline{p}_t^W - \underline{p}_f^W)^T\{(\underline{a}_t^W)^T\}^{\neg} \quad (8)$$

where $\neg$ denotes pseudo inversion.

For satisfying the normality and orthogonality of the rotation matrix, error terms are defined like the below

$$E_{Uk} = \sum_{m=1}^{3} r_{mk}^2 - 1, \qquad k = 1,2 \quad (9)$$

$$E_o = r_{11}r_{12} + r_{21}r_{22} + r_{31}r_{32} \quad (10)$$

The first and second columns of the rotation matrix can be adjusted to reduce the error terms with

$$\frac{\partial E_{orth}}{\partial r_{m1}} = 2E_{U1}r_{m1} + E_O r_{m2} \quad (11)$$

$$\frac{\partial E_{orth}}{\partial r_{m2}} = 2E_{U2}r_{m2} + E_O r_{m1} \quad (12)$$

where $E_{orth} = \frac{1}{2}(E_{U1}^2 + E_{U2}^2 + E_O^2)$. The third column can then be determined from the two columns learned by

$$\underline{r}_3 = \underline{r}_1 \times \underline{r}_2 \quad (13)$$

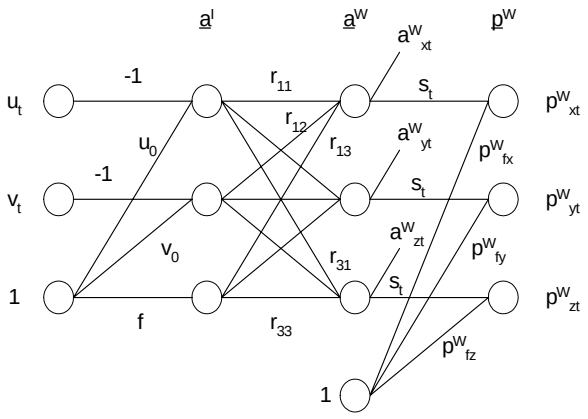

Fig.3. Neural network for camera calibration

2.3 Projection and back-projection

A ray of sight can be determined using eq.(3) when an image point is given as already described. Note that this is actually a back-projection: transformation from image to space. Thus, if two corresponding image points of stereo cameras are given, a 3D point can be specified at the intersection of the two rays. Assuming two rays and equating them leads to

$$\underline{p}_{f1}^W + s_{t1}\underline{R}_{WI1}\underline{a}_{t1}^I = \underline{p}_{f2}^W + s_{t2}\underline{R}_{WI2}\underline{a}_{t2}^I \quad (14)$$

or

$$\underline{p}_{f1}^{W} + s_{t1}\underline{a}_{t1}^{W} = \underline{p}_{f2}^{W} + s_{t2}\underline{a}_{t2}^{W} \quad (15)$$

Then, we can define a 3D point on either ray from an image point at the distance of the scale factor which can be determined from

$$\begin{bmatrix} s_{t1} \\ s_{t2} \end{bmatrix} = \left[\underline{a}_{D}^{W}\right]^{\dagger} \underline{p}_{fD}^{W} \quad (16)$$

where $\underline{a}_{D}^{W} = \begin{bmatrix} a_{x1}^{W} & -a_{x2}^{W} \\ a_{y1}^{W} & -a_{y2}^{W} \\ a_{z1}^{W} & -a_{z2}^{W} \end{bmatrix}$, $\underline{p}_{fD}^{W} = \begin{bmatrix} p_{fx2}^{W} - p_{fx1}^{W} \\ p_{fy2}^{W} - p_{fy1}^{W} \\ p_{fz2}^{W} - p_{fz1}^{W} \end{bmatrix}$

On the other hand, the reverse problem – projection; determining an image point from a 3D point given, is also can be solved by computing a relevant scale factor. Since an image point is formed at a spot where a ray from a 3D point meets the image plane, following equation is set from a 3D point given and a focal point calibrated.

$$\underline{p}_{t}^{I} = \underline{p}_{f}^{I} - \frac{1}{s_t}\underline{R}_{IW}\,(\underline{p}_{t}^{W} - \underline{p}_{f}^{W}) \quad (17)$$

Since $\underline{p}_{t}^{I} = (u, v, 0)^{T}$, the scale factor can be computed from

$$0 = f - \frac{1}{s_t}\{r_{13}(x_t - p_{fx}) + r_{23}(y_t - p_{fy}) + r_{33}(z_t - p_{fz})\} \quad (18)$$

An ANN structure for projection shown in Fig.4 has the reverse direction of propagation to that of the network in Fig.3. Of course, the weights of the network need not to be learned again if the network of Fig.3 is trained already or vice versa.

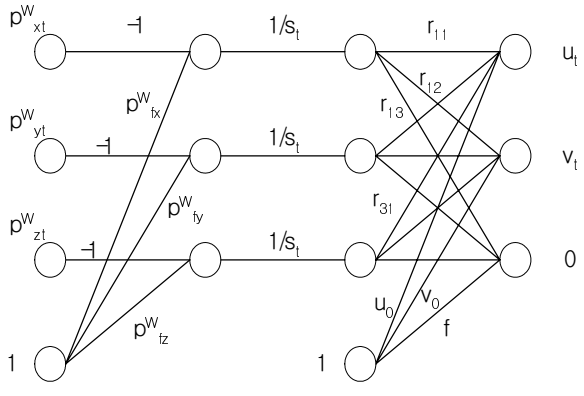

Fig.4. Neural network for projection

3. Results

A camera model in specific parameters was simulated for testing the network designed. Three sets of data were synthesized like

Data I ~ almost linear model
($k=0.00001[mm^{-2}]$)
Data II ~ significant radial lens distortion
($k=0.0001[mm^{-2}]$)
Data III ~ even non-radial lens distortion
($k=0.0001[mm^{-2}]$, $s_1=-0.0001[mm^{-1}]$, $s_2=0.0001[mm^{-1}]$, $p_1=-0.0002[mm^{-1}]$, $p_2=0.00005[mm^{-1}]$)

All data were added by zero mean Gaussian random noise of 1/5[pixel] variance. The parameters in parenthesis above are lens distortion coefficients by the Weng's model [13]. See Table 1 for the intrinsic and extrinsic camera parameters used for the test and the results obtained after 10,000 learning epochs. Only 50 data were used for the training. Fig.5 shows the plot of network learning for the three data sets. Note that the learning was fast and stable.

After completing the learning, the network was applied for the projection of the other 50 data, which were not used for learning, for generalization test. The projection errors resulted were 0.08, 0.20, and 0.28 [pixel] for Data I, II, and III respectively.

Table 1. Results of neural learning for different data

Parameters	Real values	Estimated values by ANN		
		Data I	Data II	Data III
d_x [mm]	-200.00	-199.98	-198.39	-198.47
d_y [mm]	500.00	499.99	499.12	498.94
d_z [mm]	2000.00	2000.00	1999.10	1999.00
r_{11}	0.612	0.612	0.611	0.622
r_{12}	0.047	0.047	0.042	0.039
r_{13}	0.789	0.789	0.791	0.782
r_{21}	0.612	0.613	0.613	0.607
r_{22}	-0.660	-0.660	-0.657	-0.654
r_{23}	-0.436	-0.436	-0.439	-0.450
r_{31}	0.500	0.500	0.501	0.495
r_{32}	0.750	0.750	0.753	0.755
r_{33}	-0.433	-0.442	-0.427	-0.430
u_0 [pixel]	258.00	257.59	255.96	271.02
v_0 [pixel]	204.00	203.44	196.74	191.68
f [mm]	25.00	25.00	25.00	24.99

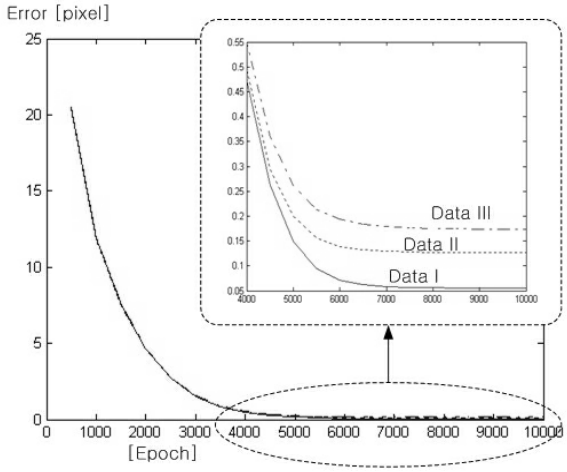

Fig.5. Network learning error

4. Conclusion

In this paper, a new camera calibration method using an artificial neural network is proposed. By using a neural network, learning parameters by optimizing the transformation between space and image plane can be done at the same time satisfying constraints on parameters such as ortho-normality of the rotation matrix. The network is designed so that an explicit calibration is feasible; each weight of synapse corresponds uniquely to one of intrinsic and extrinsic parameters of a camera calibrated. This design approach brings many practical advantages including followings:

- Once calibrated, we can use the result even after camera motion
- Calibration result by projection can be used for back-projection and vice versa
- A good network structure needs not be searched tediously for a camera
- We can assume good initial values for network's weights when we know the meaning of each network connection
- The learning is fast

A network is designed to learn a ray of sight from an image point or a 3D point given. This is different from most existing camera calibration techniques because they usually find parameters from projection transformation; finding an image point from a 3D point. Since the transformation from a point, whether it is an image point or a 3D point, to a ray of sight is one-to-one mapping, extension from the transformation is quite straightforward. Note that this kind of straightforwardness could be obtained only when we abandon one of variables available in conventional methods [14].

The proposed method can be applied in various ways. In visual inspection, for example, a 3D position can be determined at the intersection of the two rays of sight from stereo cameras. In computer graphics, an image can be determined at the point, where a ray from a 3D point meets the image plane.

Acknowledgement:

This work was supported in part by the Brain Korea 21 (BK21) project of the Korean Ministry of Education.

References:

[1] O.D.Faugueras and G.Toscani, The Calibration Problem for Stereoscopic Vision, *Sensor Devices and Systems for Robotics (A.Casals, ed.), NATO ASI Series*, Vol.F52, Springer-Verlag, Berlin, 1989, pp.195-213.

[2] G-Q.Wei and S.D.Ma, Implicit and Explicit Camera Calibration: Theory and Experiments, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.16, No.5, 1994, pp.469-480.

[3] R.Y.Tsai, A Versatile Camera Calibration Technique for High Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses, *IEEE J.Robotics & Automation*, Vol. RA-3(4), 1987, pp.323-344.

[4] M.Ito, Robot Vision Modelling - Camera Modelling and Camera Calibration, *Advanced Robotics*, Vol.5(3), 1991, pp.321-337.

[5] J.Salvi, *et al*., A Comparative Review of Camera Calibrating Methods with Accuracy Evaluation, *Pattern Recognition*, Vol.35, 2002, pp.1617-1635.

[6] M.Kume and T.Kanade, Camera System with Neural Network Compensator for Measuring 3-D Position, *U.S. Patent*, No.5617490, 1997.

[7] J.Wen and G.Schweitzer, Hybrid Calibration of CCD Cameras Using Artificial Neural Nets, *Proc. Int. Joint Conf. Neural Networks*, 1991, pp.337-342.

[8] D-H.Choi and S-Y.Oh, Real-time Neural Network Based Camera Localization and Its Extension to Mobile Robot Control," *Int. J. Neural Systems*, Vol.8(3), 1997, pp.279-293.

[9] Y.Do, Application of Neural Networks for Stereo-camera Calibration, *Proc. Int. Joint Conf. Neural Networks*, 1999, pp.2719-2722.

[10] J.Neubert, *et al*., Automatic training of a neural

net for active stereo 3D reconstruction, *Proc. IEEE Int. Conf. Robotics and Automation*, 2001, pp.2140-2146.
[11] M.Ahmed, *et. al.*, A Neural Approach for Single- and Multi-image Camera Calibration, in *Proc. Int. Conf. Image Processing*, 1999, pp.925-929.
[12] M.Ahmed and A.Farag, Locked, Unlocked and Semi-locked Network Weights for Four Different Camera Calibration Problems, *Proc. Int. Joint Conf. Neural Networks*, 2001, pp.2826-2831.
[13] J.Weng *et al.*, Camera Calibration with Distortion Models and Accuracy Evaluation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.16, No.5, 1992, pp.965-980.
[14] Y.Do *et al.*, Direct Calibration Methodology for Stereo Cameras, *Proc. SPIE Conf. Vol.3521: Machine Vision Systems for Inspection and Metrology VII*, 1998, pp.54-65.