Error Analysis of Head Pose and Gaze Direction from Stereo Vision

Rhys Newman and Alexander Zelinsky
Research School of Information Sciences and Engineering
Australian National University
Canberra ACT 0200
{newman,alex}@syseng.anu.edu.au

Abstract

There is a great deal of current interest in human face/head tracking as there are a number of interesting and useful applications. Not least among these is the goal of tracking the head in real–time[1]. An interesting extension to this problem is estimating the subject's gaze direction in addition to his/her head pose. Basic geometry shows how to perform this using the head pose and pupil locations, but it is unclear how accurate a result can be expected. In addition, many methods of computing 3D pose and results based on it are not well suited to computing real–time error estimates.

This paper describes a methodology whereby the head pose and Gaze direction, together with an estimate of their accuracy, can be computed in real–time. Preliminary numerical results are also presented.

1 Introduction

The continuing growth of consumer electronics coupled with the falling cost of computer components makes possible a number of interesting applications of computer vision. Tracking a human face automatically has been used for gesture recognition [Wu *et al.*, 1998], identification and many other applications [Rosenfeld, 1997; Chellappa *et al.*, 1995]. There has also been increasing interest in using human gestures to control *Human Friendly* robots[2].

The system described here aims to determine automatically not only the head position and orientation (i.e. *pose*), but also the direction in which the subject is looking (i.e. *gaze direction*). Two additional goals are that the system should operate at NTSC frame rate and be non-intrusive[3]. Such a system is ideal for applications where a robot is to react to human face position and gaze direction [Zelinsky and Heinzmann, 1998].

[1] For most purposes this is NTSC frame rate (30Hz).

[2] See, for example, the proceedings of the First International Workshop on Human Friendly Robotics, Tsukuba, Japan 1998.

2 System Outline

The arrangement of the stereo vision system is shown in figure 1. The two cameras are placed equidistant from the origin and are verged (angled towards the origin in the horizontal plane) at about 5^o. This is designed to offer the best measurements of an object the size of a human head placed about 600mm in front of the cameras.

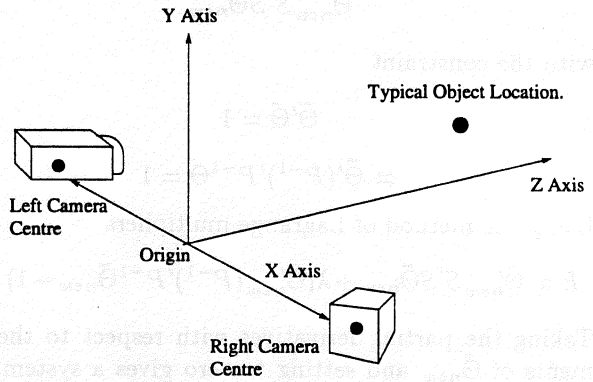

Figure 1: The Stereo Camera arrangement.

This section describes the procedure for obtaining the gaze point estimate given the two images. Section 3 describes a method for obtaining an estimate of the gaze direction error, while some initial results are discussed in section 4.

2.1 Stereo Vision

In what follows the standard pinhole camera model is used as any non–linear camera effects are secondary to the typical sources of error: noise and matching inaccuracies. A number of methods have been used to perform

[3] For this reason spcialised helmets and/or infrared lighting are not used.

stereo matching (i.e. find the image of a 3-D point in both camera planes) and such techniques will not be reviewed here. Rather the starting point for this analysis is immediately after the matching has been performed. The goal is to calculate how the geometry of the system affects the error of the head pose and gaze point estimates.

The following definitions are needed:

1. The coordinate system, termed *world coordinates*, has its origin midway between the two camera centres, with the z axis measured positive to the front of them (see figure 1).
2. The left camera centre is $\vec{c}_l$ and the right $\vec{c}_r$.
3. The image of each feature point in the left image is $\vec{x}_i^l = (u_i, v_i, 1)$ and $\vec{x}_i^r = (w_i, y_i, 1)$ in the right[4].

Typically, techniques from projective geometry are used to reconstruct the three dimensional point $\vec{x}_i$ which corresponds to $\vec{x}_i^r$ and $\vec{x}_i^l$. However these complexities are unnecessary when calibrated stereo cameras are used and can make error analysis more difficult. Instead, the camera parameters are used to transform $\vec{x}_i^r$ and $\vec{x}_i^l$ into the world coordinates directly. The results are $\vec{p}_i$ and $\vec{q}_i$ which represent directions from the camera centres, through the measured point on the image plane and out into the space in front of the camera.

$$\vec{p}_i = R_{xz}(\theta_l).R_{xy}(\phi_l).R_{yz}(\gamma_l).f_l.\vec{x}_i^l$$

$$\vec{q}_i = R_{xz}(\theta_r).R_{xy}(\phi_r).R_{yz}(\gamma_r).f_r.\vec{x}_i^r$$

Here f_l and f_r are the (scalar) focal lengths of the left and right cameras respectively, and R_{xz}, R_{xy} and R_{yz} are rotations around the y, z and x axes respectively. All rotations are included so that uncertainties may be calculated although, as shown in figure 1, $\theta_l \approx -5^0$, $\theta_r \approx 5^0$ and $\phi_l \approx \phi_r \approx \gamma_l \approx \gamma_r \approx 0$.

As $\vec{p}_i$ and $\vec{q}_i$ are corrupted by noise the three dimensional point $\vec{x}_i$ which each pair represents is sought by minimising E_i (following [Trucco and Verri, 1998]):

$$E_i = ||\vec{p}_i t_i + \vec{c}_l - \vec{x}_i||^2 + ||\vec{q}_i s_i + \vec{c}_r - \vec{x}_i||^2$$

with respect to the three coordinates of $\vec{x}_i$ and the two scalars s_i and t_i.

Setting the partial derivatives to zero yields the following solution for $\vec{x}_i$:

$$P_i = \frac{\vec{p}_i.\vec{p}_i^T}{||\vec{p}_i||^2} - \mathbf{I} \quad Q_i = \frac{\vec{q}_i.\vec{q}_i^T}{||\vec{q}_i||^2} - \mathbf{I}$$

$$(P_i + Q_i)\,\vec{x}_i = P_i.\vec{c}_l + Q_i.\vec{c}_r \qquad (1)$$

[4]The first two dimensions are the coordinates on each camera's image plane.

2.2 Recovering Head Pose

Recovery of an object's pose from observed three–dimensional data is a classical problem with a long history [Trucco and Verri, 1998; Haralick and Shapiro, 1993; Faugeras, 1993]. The problem is to determine, given a set of known *model* points $\vec{m}_i$ in three dimensions, the translation T and rotation R which place the model at the observed position/orientation. Because of the noise inherent in any observation of the transformed model, this problem is universally cast as a minimisation problem. Thus the goal is to minimise the error E with respect to R and T given the model points $\vec{m}_i$ and the observed points $\vec{x}_i$:

$$E = \sum_{i=0}^{n} ||\vec{x}_i - R\vec{m}_i - T||^2 \qquad (2)$$

Here n is the total number of points in the model.

There are two classes of solutions in the literature: iterative and direct. The first class uses initial estimates of T and R and refines them by reducing E at each step. In contrast, direct methods define a formula or procedure from which the minimum is obtained. Iterative methods have been preferred historically as they are simpler theoretically and can be made less sensitive to noise. The direct methods however, although they require solving a cubic or quartic polynomial, are more amenable to speed optimisation and error analysis. It is for this reason that a direct method, similar to Horn's result [Horn, 1987], has been developed.

For the application considered for this system, i.e. head tracking, 8 model points are used: the two corners of each eye, the corners of the mouth and the inner ends of the eyebrows. Note that it is not generally practical to extract the model ($\vec{m}_i$) by physically measuring these facial features. Rather these are extracted from a single stereo image of the subject. Consequently errors are also present in the model measurements and ways of overcoming this problem is the task of future research. For the analysis presented here it is assumed there are no errors in the model measurements ($\vec{m}_i$).

At the minimum E (equation 2) its partial derivatives with respect to the parameters which comprise R and T must vanish. The obvious representation of T will suffice but R presents a number of alternatives. A particularly useful representation for R is obtained by using a unit quarternion. The interested reader is referred to [Faugeras, 1993; Horn, 1987] for a detailed exposition of quarternions. For the purposes here it suffices to note that any rotation matrix R can be expressed using four numbers a, b, c and d as follows:

$$R = \begin{pmatrix} a^2+b^2-c^2-d^2 & 2(bc-ad) & 2(bd+ac) \\ 2(bc+ad) & a^2-b^2+c^2-d^2 & 2(cd-ab) \\ 2(bd-ac) & 2(cd+ab) & a^2-b^2-c^2+d^2 \end{pmatrix}$$

where $a^2+b^2+c^2+d^2=1$.

Before differentiating E is is helpful to expand equation[5] 2 and exploit the fact that $R^TR=\mathbf{I}$:

$$\mathcal{T}_i = \|\vec{x}_i\|^2 + \|\vec{m}_i\|^2$$

$$E = \sum_i \left(\mathcal{T}_i + \|T\|^2 - 2\vec{m}_i^T R^T(\vec{x}_i - T) - 2\vec{x}_i^T T\right) \quad (3)$$

Setting the differential with respect to T equal to zero produces a simple formula for the translation:

$$T = \frac{\sum_i \vec{x}_i}{n} - R\frac{\sum_i \vec{m}_i}{n} = \vec{\mathcal{X}} - R\vec{\mathcal{M}}$$

where $\vec{\mathcal{X}}$ and $\vec{\mathcal{M}}$ are the averages of the measurements and model points respectively. Because the origin is arbitrary $\vec{\mathcal{M}}$ can be chosen to be zero, thus leaving only one term in E which must be determined. This last term, denoted E' below, must be minimised subject to the constraint that R be a rotation matrix. The method of Lagrange is helpful here:

$$E' = -2\sum_i (\vec{x}_i - \vec{\mathcal{X}})^T R\vec{m}_i \; + \; \lambda(a^2+b^2+c^2+d^2-1) \quad (4)$$

Setting the differential of E' with respect to a, b, c and d equal to zero yields four linear equations:

$$\sum_i (\vec{x}_i - \vec{\mathcal{X}})^T \begin{pmatrix} a & -d & c \\ d & a & -b \\ -c & b & a \end{pmatrix} \vec{m}_i - \lambda a = 0$$

$$\sum_i (\vec{x}_i - \vec{\mathcal{X}})^T \begin{pmatrix} b & c & d \\ c & -b & -a \\ d & a & -b \end{pmatrix} \vec{m}_i - \lambda b = 0$$

$$\sum_i (\vec{x}_i - \vec{\mathcal{X}})^T \begin{pmatrix} -c & b & a \\ b & c & d \\ -a & d & -c \end{pmatrix} \vec{m}_i - \lambda c = 0$$

$$\sum_i (\vec{x}_i - \vec{\mathcal{X}})^T \begin{pmatrix} -d & -a & b \\ a & -d & c \\ b & c & d \end{pmatrix} \vec{m}_i - \lambda d = 0$$

These can be collected into a matrix $\mathcal{A}$ and written:

$$(\mathcal{A} - \lambda\mathbf{I})\vec{s} = 0 \quad (5)$$

where $\vec{s}^T = (a,b,c,d)^T$

Thus the non-trivial solutions are the eigenvectors of $\mathcal{A}$ and, as shown in [Horn, 1987], the choice which minimises E' is the eigenvector which corresponds to the maximum eigenvalue. There are a number of methods to obtain the eigenvectors of a matrix, however as the characteristic polynomial is only quartic, a direct solution method such as Ferrari's [Uspensky, 1948] can be employed. This is especially important when the solution is requried in real–time.

[5]The sum $\sum_{i=0}^{n}$ has been abbreviated to $\sum_i$.

2.3 Gaze Point/Direction Estimation

The relationship between the eye centres, pupils and gaze point is exactly analogous to that between the camera centres, an object's image in each camera and the object's 3D position. Note that the offset between the model points (section 2.2) and the eye centres are constant. Therefore if, in addition to the locations of the visible facial features, the model contains these offsets the eye centres can be found from the known pose (previous section). Again it is assumed these offsets contain no error in themselves, but the eye centres suffer inaccuracies from the pose estimation procedure.

Leaving to one side the issue of how the centre of the pupils are detected in both images, their 3D location can be computed in the manner described in section 2.1. Deonte the 3D results of these calculations as follows:

1. $\vec{e}_l$ the position of the left pupil.
2. $\vec{e}_r$ the position of the right pupil.
3. $\vec{r}$ the position of the right eye centre.
4. $\vec{l}$ the position of the left eye centre.
5. $\vec{d}_l = \vec{e}_l - \vec{l}$
6. $\vec{d}_r = \vec{e}_r - \vec{r}$

The 3D gaze point $\vec{G}$ is then given by equation 1:

$$D_l = \frac{\vec{d}_l.\vec{d}_l^T}{\|\vec{d}_l\|^2} - \mathbf{I} \quad D_r = \frac{\vec{d}_r.\vec{d}_r^T}{\|\vec{d}_r\|^2} - \mathbf{I}$$

$$(D_l + D_r)\,\vec{G} = D_l\vec{l} + D_r\vec{r}$$

To continue, define $\vec{g}$:

$$\vec{g} = \vec{G} - \frac{\vec{r}+\vec{l}}{2}$$

The Gaze direction $\vec{\mathcal{G}}$ is therefore given by a two dimensional vector which contains the tangents of the angles formed by the gaze line with the z axis:

$$\vec{\mathcal{G}} = \begin{pmatrix} \frac{\vec{g}_x}{\vec{g}_z} \\ \frac{\vec{g}_y}{\vec{g}_z} \end{pmatrix}$$

3 Error Analysis

This section analyses the errors introduced from a number of sources into the three stages of calculation presented in the previous section. Following the standard local linearity assumption [Faugeras, 1993], if an m dimensional vector $\vec{x}$ suffers errors in its components which are charaterised by an $m\times m$ covariance matrix Λ, a function $F:\mathbf{R}^m \rightarrow \mathbf{R}^n$ of $\vec{x}$ will suffer errors charaterised by the $n \times n$ covariance matrix Λ':

$$\Lambda' = DF(\vec{x})\;\Lambda\;DF(\vec{x})^T \quad (6)$$

Here $DF(\vec{x})$ is the $n \times m$ differential matrix of F evaluated at $\vec{x}$. In short, the errors in the output are related to those of the input by a linear transformation. Combining the transformations of various stages together yields an accuracy estimate of the final result[6].

3.1 Stereo Reconstruction

The construction of a 3D point from two views depends not only on the measurements but also on the positions and aspect of the cameras. Thus equation 1 is thought of as a function from $F : \mathbf{R}^{18} \rightarrow \mathbf{R}^3$ where the domain includes 7 camera parameters for each camera as well as the measurements themselves:

$$\vec{x}_i = F(\theta_r, \theta_l, \phi_r, \phi_l, \gamma_r, \gamma_l, f_r, f_l, \vec{c}_r, \vec{c}_l, \vec{x}_i^r, \vec{x}_i^l) \qquad (7)$$

The differential of F is a complicated 3×18 matrix which is best found using a computer algebra package. In practise the computation is simplified significantly by obtaining the derivative of $\vec{x}_i$ from the differential of equation 1:

$$\left(\frac{\partial P_i}{\partial \omega} + \frac{\partial Q_i}{\partial \omega}\right)\vec{x}_i + (P_i + Q_i)\frac{\partial \vec{x}_i}{\partial \omega} = \frac{\partial\left(P_i.\vec{c}_l + Q_i.\vec{c}_r\right)}{\partial \omega}$$

Here ω is any one of the domain variables of F.

The 3×1 vector $\frac{\partial \vec{x}_i}{\partial \omega}$ forms a column of $DF(\vec{x}_i)$ depending on which of the 18 domain variables ω is. The covariance matrix of the 3D position is then calculated using equation 6 where Λ is assumed to be diagonal: i.e. the initial errors in each variable are independent.

3.2 Pose Estimation

The most difficult part of estimating the pose error is determining a value for $\frac{\partial \vec{s}}{\partial \omega}$ (where ω is one of the domain variables of F). Again this can be found by differentiating equation 5:

$$\left(\frac{\partial \mathcal{A}}{\partial \omega} - \frac{\partial \lambda}{\partial \omega}\mathbf{I}\right)\vec{s} + (\mathcal{A} - \lambda\mathbf{I})\frac{\partial \vec{s}}{\partial \omega} = 0$$

It is a simple exercise to verify that $\mathcal{A}$ is symmetric, so by pre-multiplying both sides by $\vec{s}^T$ a formula is obtained for the differential of λ:

$$\frac{\partial \lambda}{\partial \omega} = \vec{s}^T \frac{\partial \mathcal{A}}{\partial \omega}\vec{s}$$

Combining these two produces:

$$(\mathcal{A} - \lambda\mathbf{I})\frac{\partial \vec{s}}{\partial \omega} = (\vec{s}\vec{s}^T - \mathbf{I})\frac{\partial \mathcal{A}}{\partial \omega}\vec{s} \qquad (8)$$

This equation is not sufficient as $(\mathcal{A} - \lambda\mathbf{I})$ is singular; its null space includes at least the space of vectors spanned by $\vec{s}$. Fortunately an additional equation can be obtained from the constraint $\|\vec{s}\| = 1$:

$$\vec{s}^T \frac{\partial \vec{s}}{\partial \omega} = 0$$

In theory it is possible for the nullity of $(\mathcal{A} - \lambda\mathbf{I})$ to be greater than one when the largest eigenvalue of $\mathcal{A}$ is repeated. However this situation corresponds to there being two rotations R which minimise the error E (see equation 4) and as such is virtually impossible[7]. Therefore combining equation 8 with the constraint that $\|s\| = 1$ is almost always sufficient to determine the derivative of $\vec{s}$.

3.3 Gaze Direction Estimation

As noted in section 2.3 the calculation of the gaze point proceeds in an identical way the 3D data points $\vec{x}_i$ are obtained from the image data points. The error analysis technique is therefore identical to that outlined in section 3.1.

4 Experimental Results

To determine the typical accuracies which can be expected from the stereo system described in section 1 some numerical results were obtained. A Monte–Carlo simulation was conducted where independent Gaussian noise $\mathcal{N}(0, \sigma^2)$ was added during various parts of the gaze point calculation:

Variable	σ
Pixel Position	0.1mm ($\pm$2 Pixels)
Camera Position	1mm
3 Camera Rotation angles	2^o
Focal Length	2mm

The variation in the camera rotation angles and position is expected to be caused by vibration. Thus the error is not slowly varying (such as temperature drift) and cannot be accounted for simply by re–calibration. The pixel uncertainty represents any error introduced by the matching algorithm used. Aside from these random components, the focal length was 30mm, the camera centres placed at $\pm$60mm and the vergence angle was 5^o.

Each simulation consisted of the following steps:

1. Choose a random head pose and gaze direction:
   (a) The head orientation is directly towards the cameras with uniformly random rotation angles added $\in [-30^o, 30^o]$.
   (b) The head position is chosen uniformly from within $x, y \in [-100, 100]$, $z \in [200, 400]$.

[6] In this case the gaze point.

[7] Either the model points ($\vec{m}_i$) are degenerate in some way or the noise makes them appear so. The face model used does not present these difficulties.

(c) The gaze position is chosen uniformly from within a box 300mm large, centred 300mm in front of the head position.

2. Perform the following 200 times:

   (a) Compute the locations of the model points in each image.

   (b) Add the noise.

   (c) Compute the gaze direction. Store these in an array Y.

3. Compute the covariance matrix of the gaze direction results stored in Y.

This procedure was repeated 200 times and the average gaze direction error was 1.566. Thus, on average, the gaze direction measurement was corrupted by Gaussian noise with $\sigma = 1.566$. This translates into an error of about 60^o.

It is worth noting that this large error is only present for a few (but significant) choices of head pose/gaze position[8]. When the subject is directly in front of the cameras and looking straight ahead, the error is significantly less: $\sigma = 0.0425$. This shows why it is important to compute the error during operation rather than fix it at some pre-computed value.

When vibration is less of a problem, the errors in the camera position and attitude will be considerably reduced. With this in mind a second set of results was compiled under the assumptions below:

Variable	σ
Pixel Position	0.05mm (+/- 1 Pixels)
Camera Position	0.31mm
3 Camera Rotation angles	1^o
Focal Length	1mm

The gaze direction results in this case were corruped by Gaussian noise with $\sigma = 0.1099$. This corresponds to about 6^o over all head positions/gaze directions[9].

An approximation algorithm for the error was also developed using the techniques outlined in section 3. As expected[10] the predictions of this method are not as accurate as the those of the Monte–Carlo simulation. It does, however, reflect the quality of the gaze point: if the error is small then its computation of the covariance matrix is small too (and vice–versa). More importantly however it is possible to compute the linearised approximation to the covariance matrix in real–time: the current version[11] takes 47ms as opposed to the Monte–Carlo method which takes over 10 seconds.

[8] Indeed the maximum σ in this set was 143.

[9] The maximum σ in this set was 0.3244.

[10] Non–linear terms are not accounted for.

[11] Optimisations will improve this still further.

5 Conclusion

This paper assesses the potential of a stereo camera system to measure the head pose and gaze direction of a human subject whose head is approximately 600mm in front of the cameras. The results presented show that, given careful but achievable camera calibration and vision tracking, it is possible to estimate the gaze direction to within 6^o degrees (on average).

The algorithm described is a simple and efficient solution to the $3D - 3D$ pose problem which is at the heart of the Gaze direction calculation. Additional results are also presented which show how to compute efficiently an estimate of the Gaze direction's error. Having such an estimate available during real–time operation facilitates a significant performance improvement to other systems which either do not estimate the error, or have to calculate it off–line. A number applications of this system are currently being investigated including human/computer interaction, consumer research and ergonomic assessment.

References

[Chellappa *et al.*, 1995] R Chellappa, C Wilson, and S Sirohey. Humand and machine recognition of faces: a survey. *Proceedings of the IEEE*, 83(5):705–740, 1995.

[Faugeras, 1993] O. Faugeras. *Three Dimensional Computer Vision: A Geometric Viewpoint.* MIT Press, 1993.

[Haralick and Shapiro, 1993] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*, volume 2. Prentice Hall, 1993.

[Horn, 1987] B.K.P. Horn. Closed–form solution of absolute orientation using unit quarternions. *Journal of the Optical Society of America: A*, 4(4):629–642, 1987.

[Rosenfeld, 1997] A. Rosenfeld. Survey: image analysis and computer vision 1996. *CVGIP: Image Understanding*, 66(1):33–93, 1997.

[Trucco and Verri, 1998] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision.* Prentice Hall, 1998.

[Uspensky, 1948] J. V. Uspensky. *Theory of Equations.* McGraw-Hill, 1948.

[Wu *et al.*, 1998] H W Wu, T Shioyama, and H Kobayashi. Spotting recognition of head gestures from colour image series. In A K Jain, S Venkatesh, and B C Lovell, editors, *ICPR Proceedings 1998*, pages 83–85, 1998.

[Zelinsky and Heinzmann, 1998] A Zelinsky and J Heinzmann. A novel visual interface for human-robot communication. *Advanced Robotics*, 11(8):827–852, 1998.