International Journal of Computer Vision 68(3), 219–237, 2006

DOI: 10.1007/s11263-005-5032-8

Automated Alignment of Robotic Pan-Tilt Camera Units Using Vision

JOSS KNIGHT AND IAN REID
Active Vision Lab, Robotics Research Group, Department of Engineering Science, University of Oxford, Parks Road, Oxford OX1 3PJ, UK
joss@robots.ox.ac.uk
ian@robots.ox.ac.uk

Received March 3, 2004; Revised June 1, 2005; Accepted June 14, 2005

First online version published in May, 2006

Abstract. In this paper we show how to carry out an automatic alignment of a pan-tilt camera platform with its natural coordinate frame, using only images obtained from the cameras during controlled motion of the unit. An active camera in aligned orientation represents the zero position for each axis, and allows axis odometry to be referred to a fixed reference frame; such referral is otherwise only possible using mechanical means, such as end-stops, which cannot take account of the unknown relationship between the camera coordinate frame and its mounting. The algorithms presented involve the calculation of two-view transformations (homographies or epipolar geometry) between pairs of images related by controlled rotation about individual head axes. From these relationships, which can be calculated linearly or optimised iteratively, an invariant line to the motion can be extracted which represents an aligned viewing direction. We present methods for general and degenerate motion (translating or non-translating), and general and degenerate scenes (non-planar and planar, but otherwise unknown), which do not require knowledge of the camera calibration, and are resistant to lens distortion non-linearity.Detailed experimentation in simulation, and in real scenes, demonstrate the speed, accuracy, and robustness of the methods, with the advantages of applicability to a wide range circumstances and no need to involve calibration objects or complex motions. Accuracy of within half a degree can be achieved with a single motion, and we also show how to improve on this by incorporating images from further motions, using a natural extension of the basic algorithm.

Keywords: alignment, active vision, calibration

1. Introduction

An active camera is any camera with robotic control of its position or orientation, the most common variety being a camera mounted on a pan-tilt unit or 'head'. Such devices are becoming increasingly common, particularly for use as surveillance devices. Often these cameras are used to measure the bearing to an object in the field of view of the camera. The motivation behind our work is the use of stereo cameras on a navigating robot to triangulate the position of scene objects for localisation (Davison, 1998; Knight, 2002). However there are other applications, such as in steering control where the bearing to an object can be used as an input to a control law (Murray et al., 1996, 1997); or tracking applications in which the camera must at least be capable of relating an image distance to a rotation angle, and be able to rotate by that angle with some accuracy in order to track an object. In most of these situations the active camera platform, or head, will need some frame of reference for its motion. It will need to be able to answer questions such as "when am I facing forwards?" or "when am I horizontal?". This is the problem of alignment, which, put simply, is the process of establishing the angular position of each axis with respect to some fixed origin. When this is done,

Figure 1. The Yorick stereo pan-tilt head (Sharkey et al., 1993), and robot GTI, used for visual navigation.

image measurements can be referred back to a coordinate system fixed in the head rather than moving with the camera.

One way to tackle alignment is to use some kind of absolute odometry system, which will usually take the form of end stops or switches from which the origin can be calculated repeatably. However, many heads, including those used in our navigation system (see Fig. 1), do not have such guides, the zero position being stored in memory and therefore volatile. In this sort of head other ways must be found to calculate the origin of the head's coordinate system. Another motivation to find an alternative method is that switches and stops are very specific to the hardware. It is preferable to find a general algorithm that can be used regardless of the exact hardware setup.

One might instead devise a way to align the head manually, perhaps by eye, or using special measuring devices. Not only may this be of dubious accuracy, but a regular manual alignment may be tedious, or even impossible if the head is being operated remotely.

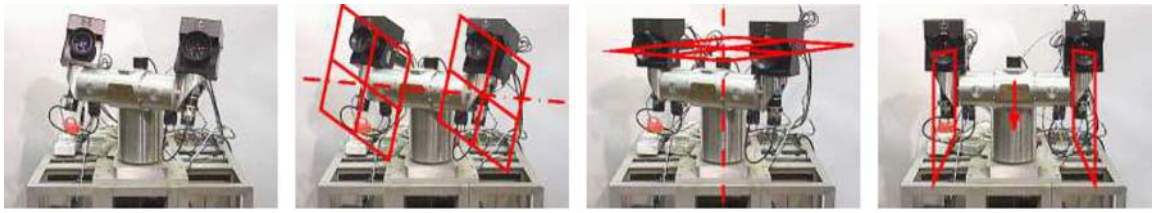
Both these methods, end-stops and manual alignment, suffer from the additional deficiency that they work relative to the camera's mechanical position, not its internal coordinate system. Accurate alignment involves moving the camera coordinate frame to a fixed position, and that frame is defined by a camera centre and optic axis which can only be approximately determined from external measurement of the camera.

In this paper we present instead a fully automatic method which uses images as its input. By making controlled motions of any angle (known or unknown) about individual axes, we can calculate aligned directions as the horizon lines of planes defined by axis orientations. This procedure can be carried out remotely and will align the camera's true coordinate frame, yet places very few restrictions on the scene being viewed, as well as being inherently a single camera solution.

1.1. A Definition of Alignment

The choice of origin for the aligned frame is primarily of importance to the extent that it must be repeatably obtainable with or without odometry via encoders. The standard definition used here is the 'natural' frame in which all head axes that control gaze direction lie perpendicular to the principal direction of the camera.[1] The reasoning of this paper is quite general and applicable to most conceivable active camera configurations, whether monocular or multi-ocular. For demonstrations and simulations, however, we have used our binocular head Yorick (Sharkey et al., 1993) as the example. Fig. 2, for instance, illustrates what alignment means in Yorick's case; the annotations show planes parallel to the direction in which the cameras must face to zero each axis.

Figure 2's description of an aligned position is of course peculiar to Yorick's pan-tilt-verge[2] arrangement



(a) Misaligned (b) Vergence axes aligned (c) Elevation axis aligned (d) Pan axis aligned

Figure 2. Aligning Yorick: in the zero position of the head's natural coordinate frame, the vergence axes are set so that the cameras face perpendicular to the elevation/tilt axis (b); the elevation axis is aligned so the cameras point perpendicular to the pan axis (c); and the pan axis is aligned so the cameras face parallel to the robot's forward direction (d). Aligning the pan axis (d) is application-specific, and not covered by this paper.

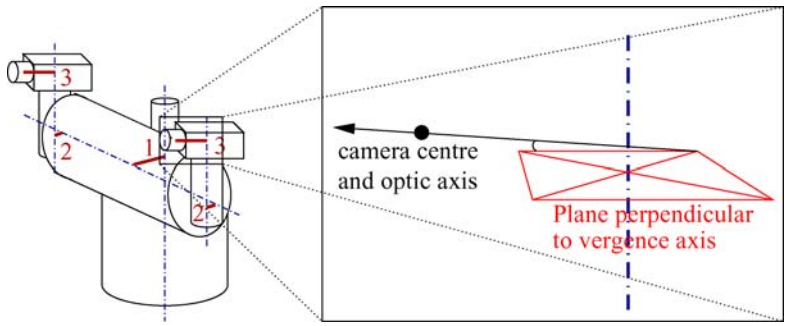


Figure 3. A diagram of the Yorick stereo head showing the offsets between the axes, and between the camera centres and axes, that make the head non-ideal. In addition, not only will the camera centres not lie on the vergence axes, but there is no guarantee that the optic axes will pass through them or lie perpendicular to them. In an ideal head this would be the case, and 1, 2 and 3 would be of zero length.

of the kinematic chain, with pan being first in the chain and independent of other axes. This configuration is typical, however others can be accounted for with similar reasoning. Using the algorithm described in this work, any axis not first in the kinematic chain can be aligned as long as its zero position is a direction perpendicular to an axis earlier in the chain. So a monocular camera platform in pan-tilt configuration can align its tilt axis; in a tilt-pan configuration it can only align the pan axis. Other clues related to the hardware setup must be used to align the first axis in the chain. In this work we therefore look only at the general problem of alignment of the other axes.

1.2. Previous Work

Alignment is a subset of the process of kinematic calibration. This is the process of calculating the pose (position and orientation) of the cameras and the axes of an active head. As illustrated by Fig. 3, the kinematics of a typical head may include unforeseen offsets. The only real guarantee is that consecutive axes are perpendicular, which is a basic design constraint for most heads.

The important kinematic parameters for our purposes are the alignment, which allows the gaze directions of the cameras to be determined, and the baseline, which effectively defines the metric scale for triangulation. This is because in our navigation application the offsets between consecutive axes can feasibly be incorporated into an overall measurement error, whereas errors in the alignment and baseline lead to a consistent bias.

Determining the full kinematics is not addressed directly in this paper, since it must be obtained via the process of camera calibration, calculating the pose of a camera from an image of a calibration object with known metric structure (classical calibration), or from multiple images of an unknown, but rigid object (self-calibration). This is in fact how kinematic calibration has been traditionally obtained, by calibrating the camera before and after rotations about each axis, which will give the relative position of that axis. Of the work on image-based kinematic calibration, (Li, 1998), and (DeSouza et al., 2002) made use of a calibration object of known metric structure, with only (Ma, 1996) opting for self-calibration based on multi-view relationships.

To test the validity of the calibration, these works compared rotation angles provided by odometry to those predicted by the kinematic model. Li (1998) points out that this kind of kinematic calibration is only valid in the ranges of motions where the calibration is performed, and this motion must be small to ensure that the calibration object remains in the field of view. Consequently, even a small error could represent several degrees of error in the axis orientations, since this would not cause a large offset for a small motion. Unfortunately, it is the orientations of the axes, not their positions, that are crucial for alignment.

McLauchlan and Murray (1996) implemented a sequential calibration algorithm for a monocular head which operated in any scene using inter-image relationships, with rotation angles taken from head encoders and therefore known (unlike in our algorithm). They simplified the problem by assuming the head axes passed through a single point (i.e. it was ideal), and over long runs achieved convergence to axis orientations with standard deviations of less than a degree.

Hayman et al. (2000) used an alternative method involving making the assumption that the head makes pure rotations about the camera centres.

A self-calibration routine provides rotation matrices representing the pose of the camera for each image in a sequence. From these the aligned position can be extracted. The process boils down to minimising the cyclotorsion component of camera motion, which will be zero for an aligned head since all axes lie perpendicular to the camera's optical axis. Hayman reports errors of between 0.5° and 1° for sequences of 30 images.

All these methods extract a full kinematic calibration rather than the alignment information alone. Instead, we present a method that is based on a strong understanding of the visual geometry of the problem. This will be shown to be simple, extract only the required information, and fast in comparison to other methods. The basic concept, that of determining the images of lines invariant to head rotations, was originally put forward by Reid and Beardsley (1996). This method required non-planar scene structure, stereo cameras, and some computationally intensive computer vision such as matching over four views and a 3D reconstruction of the scene being viewed. We show that the underlying geometry can be extracted in a considerably faster and more accurate way using two-view transformations relating images from a single camera. Our results extend, improve upon, and evaluate more thoroughly ideas we originally published in Knight and Reid (2000b).

2. The Algorithm and Geometrical Motivation

This paper requires the reader to have a basic understanding of projective geometry and multi-view relationships, including epipolar geometry and homographies. For more on these subjects, the reader is referred to Hartley and Zisserman (2000), from which we also take our notational conventions. This includes the use of bold face for vectors ($\mathbf{x}$, Π), and teletype for matrices (H, K).

The algorithms we present rely on two premises: the scene being viewed by the cameras is rigid, and the motion of the cameras can be controlled (meaning only that axes can be rotated individually, not that rotations of known angles can be made). Of particular importance is the ability to carry out a *planar motion*. Planar motion has received special attention in the literature (Armstrong, 1996; Hartley and Zisserman, 2000; Beardsley and Zisserman, 1995; Zisserman et al., 1995; Horaud and Csurka, 1998) since it commonly occurs when using mobile robots, which make motions along the ground plane (Fig. 4(a)), and active heads, which execute a planar motion when rotating about a single axis (Fig. 4(b)). Naturally it is the latter that is of interest to us.

A planar motion is defined as that for which the path of any point lies in a single plane. An alternative is to view any rigid body motion as rotation around, and translation along, some axis in space, the *screw axis*. A planar motion is one for which the translation, or pitch of the screw, is zero. It is one of the invariants to a planar motion, a fixed image line, that provides the direction for aligning head axes.

2.1. Invariants to Planar Motion

Figure 5(a) shows the 3-space invariants to planar motion, namely:

- The screw axis, and every point on it.
- The set (or 'pencil') of planes perpendicular to the screw axis
- The line on the plane at infinity that is the intersection of this pencil of planes, $\mathbf{L}$.

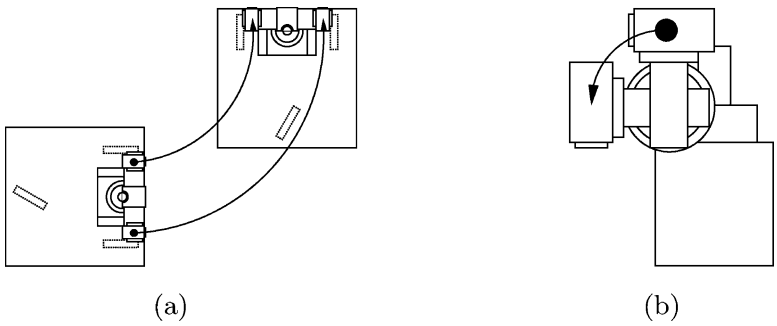


Figure 4. Examples of cameras undergoing planar motion.

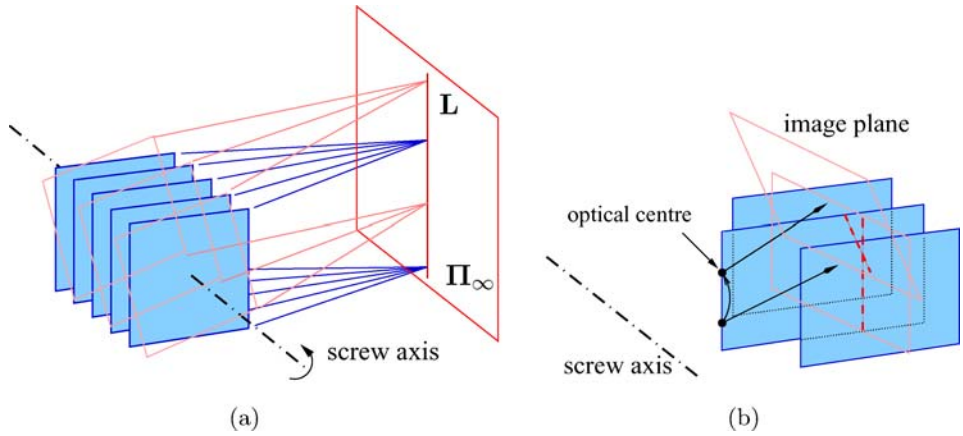


Figure 5. The (a) 3D and (b) image plane invariants to planar motion. In (b) the camera is represented as an optical centre and image plane, with the dashed line indicating the image of the invariant plane that passes through the optical centres (i.e. its intersection with the image plane). The optical axis, as in this case, need not lie in the invariant plane.

The images of some of these entities can be calculated from two-view relationships, namely:

- A line, $\mathbf{l}_s$, the image of the screw axis. This is not a line of fixed points or a fixed line, merely the image of a line of fixed points.
- A fixed line, $\mathbf{l}_h$, the image of the invariant line at infinity $\mathbf{L}$ (Fig. 5(b)—the subscript 'h' is used by analogy with the horizon line). Alternatively, it can be seen as the image of the invariant plane that passes through the optical centre.

Figure 5 shows that during planar motion the optical centre lies on one of the invariant planes, moving through it, which is why that plane is imaged as an invariant line. This also means that $\mathbf{l}_h$ is an epipolar line.

These entities are entirely a result of the nature of the motion, and therefore exist regardless of the scene being imaged. The scene may, however, affect which of the invariants can be calculated. One special case motion is if the screw axis passes through the optical centre, as is the case for rotations of the camera about its centre (referred to here as 'pure rotation'). In this case the screw axis is imaged in a fixed point $\mathbf{x}_v$, the intersection of the screw axis with the image plane, not a line. Figure 6 shows how the invariants might appear in the image in each case.

2.2. *Geometric Intuition for Alignment*

With the basic geometry understood the intuition for a general alignment algorithm follows. The procedure to align an axis A, which directly follows an axis B in the kinematic chain, is to fixate the line at infinity invariant to a motion about axis B. As Fig. 7 demonstrates, if the camera's principal direction is moved so that it passes through this line at infinity (i.e.the line is fixated), that principal direction will now lie in the pencil of planes invariant to the motion, in other words the camera is pointing perpendicular to the rotation axis, as required for alignment. This can be done for each axis to be aligned in turn, starting with the last axis in the kinematic chain if there are more than two (so, for instance, it is no good aligning the pan axis if the vergence axis is not already aligned).

The 'principal direction' of the camera need not necessarily be the optic axis (although it would ideally be) because the requirement is only that the calculation be repeatable from any initial position. Therefore 'fixation' in our case means moving the target to the image centre.[3] A general alignment algorithm could be defined as follows:

Algorithm 1. Generic Alignment Algorithm. To move head axis A to an aligned position perpendicular to axis B, move the camera about axis B alone. Based on images from this motion, calculate the image of the line at infinity invariant to the motion, and fixate any point on it.

If we wish to align an axis of cyclorotation then this can be done simultaneously with respect to head axis B by adjusting the definition of 'fixation' to include the rotation of the image plane to ensure the calculated

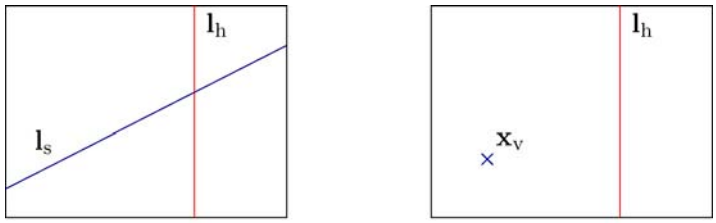


Figure 6. Invariants to planar motion as they might be seen in the image (a) in the general case, and (b) if the motion is pure rotation. $\mathbf{x}_v$ will generally not be seen in the image (for a camera with a standard field of view) unless the motion involves considerable cyclotorsion (rotation about the optic axis, i. e. the screw axis is directed towards the scene).

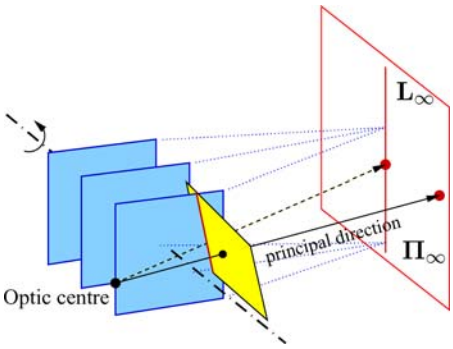


Figure 7. Illustrating the process of alignment. If the principal direction is moved so as to coincide with the invariant line at infinity, $\mathbf{L}_\infty$ (it is fixated), then it will lie in an invariant plane. Consequently the camera must be directed perpendicular to the axis of rotation, as required.

horizon lies at the required orientation, passing through the image centre, and either along or perpendicular to an image scan line depending on whether axis B is a pan or tilt (or other) axis. Again, any succeeding axes in the kinematic chain that affect cyclotorsion must also be aligned to avoid ambiguity.

In Reid and Beardsley (1996), Reid and Beardsley showed how to obtain the necessary fixation lines by calculating a 3D homography in projective 3-space. However, we can now show that the invariant line can be obtained from the 2D relationship between images taken during the motion about a single axis, avoiding the necessity of calculating projective structure, and the consequent restriction that the scene must be non-planar (and that there must be two cameras). Not only does this improve accuracy, it is considerably faster.

2.3. The Planar Fundamental Matrix

A fundamental matrix $\mathtt{F}$ can be divided into its symmetric and antisymmetric parts,

$$\mathtt{F} = \mathtt{F}_a + \mathtt{F}_s$$

where

$$\mathtt{F}_a = \frac{1}{2}(\mathtt{F} - \mathtt{F}^\top), \qquad \mathtt{F}_s = \frac{1}{2}(\mathtt{F} + \mathtt{F}^\top) \ ,$$

although the factors of a half are irrelevant since $\mathtt{F}$ has arbitrary scale. $\mathtt{F}_s$ is a conic, and $\mathtt{F}_a$ is the skew-symmetric matrix whose nullspace is a point $\mathbf{x}_a$ (i.e. $\mathtt{F}_a = [\mathbf{x}_a]_\times$). Both $\mathtt{F}_s$ and $\mathbf{x}_a$ have geometric interpretations (Hartley and Zisserman, 2000).

For a planar motion, $\mathtt{F}_s$ is degenerate, and represents the lines $\mathbf{l}_h$ and $\mathbf{l}_s$ (Fig. 6). From the form of such a degenerate conic,

$$\mathtt{F}_s = \mathbf{l}_h\mathbf{l}_s^\top + \mathbf{l}_s\mathbf{l}_h^\top \ . \tag{1}$$

For a general motion $\mathtt{F}_s$ has full rank but for planar motion it drops to rank 2. Since both $\mathtt{F}$ and $\mathtt{F}_s$ are rank 2, $\mathtt{F}$ now has only 6 degrees of freedom. $\mathbf{x}_a$ lies on $\mathbf{l}_h$ and so it provides only one degree of freedom. The other is contained in a variable θ related to the angle between views. Vieville and Lingrand gave a minimal parameterisation of a planar $\mathtt{F}$ matrix as Viéville and Lingrand (1996)

$$\mathtt{F} = \sin(\theta)[\mathbf{x}_a]_\times + (1 - \cos(\theta))[\mathbf{l}_h\mathbf{l}_s^\top + \mathbf{l}_s\mathbf{l}_h^\top], \tag{2}$$

where the terms $\mathtt{F}_a$ and $\mathtt{F}_s$ can be clearly seen along with their weights, $\sin(\theta)$ and $(1 - \cos(\theta))$.

A simpler parameterisation, described by Hartley and Zisserman (2000), comes from the epipoles $\mathbf{e}$ and $\mathbf{e}'$. Figure 8 illustrates: a point $\mathbf{x}$ in the first image lies on epipolar line $\mathbf{l}$ where $\mathbf{l} = \mathbf{e} \times \mathbf{x}$. This line intersects $\mathbf{l}_s$ at a point $\mathbf{p} = \mathbf{l}_s \times \mathbf{l}$, which must also lie on the corresponding epipolar line because $\mathbf{l}_s$ is the image of a line of fixed points. Therefore we find $\mathbf{l}' = \mathbf{e}' \times \mathbf{p} = \mathbf{e}' \times (\mathbf{l}_s \times (\mathbf{e} \times \mathbf{x}))$, or

$$\begin{aligned} \mathbf{l}' &= [\mathbf{e}']_\times[\mathbf{l}_s]_\times[\mathbf{e}]_\times\mathbf{x} \\ \Rightarrow \mathtt{F} &= [\mathbf{e}']_\times[\mathbf{l}_s]_\times[\mathbf{e}]_\times \ . \end{aligned} \tag{3}$$

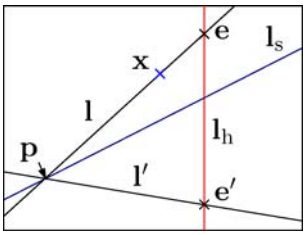


Figure 8. Constructing epipolar lines using the fixed lines $\mathbf{l}_s$, $\mathbf{l}_h$, and the epipoles. Extend the epipolar line $\mathbf{l}$, which intersects the point $\mathbf{x}$ and the epipole $\mathbf{e}$, to meet $\mathbf{l}_s$. This intersection, $\mathbf{p}$, is the image of a fixed point (as are all points on $\mathbf{l}_s$), and so also lies on the corresponding epipolar line $\mathbf{l}'$.

Since $\mathtt{F}_s$ is rank 2 it has just 2 non-zero eigenvalues: λ_0, which is negative, and λ_1, positive, with corresponding eigenvectors $\mathbf{v}_0$ and $\mathbf{v}_1$. So $\mathtt{F}_s = \mathbf{v}_0\lambda_0\mathbf{v}_0^\top + \mathbf{v}_1\lambda_1\mathbf{v}_1^\top$. By examining this and Eq. (1) it can be shown that

$$\begin{aligned}\mathbf{f}_1 &= \sqrt{\lambda_1}\cdot\mathbf{v}_1 + \sqrt{-\lambda_0}\cdot\mathbf{v}_0,\\ \mathbf{f}_2 &= \sqrt{\lambda_1}\cdot\mathbf{v}_1 - \sqrt{-\lambda_0}\cdot\mathbf{v}_0,\end{aligned} \tag{4}$$

where $\mathbf{f}_1$ and $\mathbf{f}_2$ are $\mathbf{l}_h$ and $\mathbf{l}_s$, but which is which has not yet been established.

In order to distinguish the fixed lines, the proximity of both epipoles to each could be tested, since the epipoles lie on $\mathbf{l}_h$. In this work, however, the epipoles are often poorly determined due to near-degenerate conditions (the motion is close to a pure rotation for which the epipoles are undefined). Instead, a method not sensitive to the location of epipoles was used: find the intersection of $\mathbf{f}_1$ and $\mathbf{f}_2$, $\mathbf{x}_p = \mathbf{f}_1 \times \mathbf{f}_2$. Then use $\mathtt{F}$ to find its epipolar line. Whichever of $\mathbf{f}_1$ and $\mathbf{f}_2$ most closely matches this line must be $\mathbf{l}_h$. This is done algebraically (*ie.* $\|\bar{\mathbf{f}}_1 - \overline{\mathtt{F}\mathbf{x}}_p\|$ is compared to $\|\bar{\mathbf{f}}_2 - \overline{\mathtt{F}\mathbf{x}}_p\|$, the bar indicating use of normalised vectors)[4].

2.4. *The Planar Motion 2D Homography*

Since a homography is a simple one-to-one mapping, its fixed entities can be found trivially from its eigenvalues. A homography $\mathtt{H}$ must have either one or three real eigenvalues, but it will only have three for certain special conditions. Thus the eigenvector associated with the real eigenvalue of the point transfer homography $\mathtt{H}$ is the fixed point $\mathbf{x}_v$ (see Fig. 6). The dual of this transformation is the line transfer homography $\mathtt{H}^{-\top}$. The real eigenvector of this matrix is the fixed line $\mathbf{l}_h$.

Equivalently, the two complex conjugate eigenvectors of $\mathtt{H}$ are complex invariant points that lie on $\mathbf{l}_h$. Their real and imaginary parts (their sum and difference) are real points that lie on $\mathbf{l}_h$, so the intersection of these can be used to find it. Similarly the complex conjugate eigenvectors of $\mathtt{H}^{-\top}$ are lines that intersect at the invariant point $\mathbf{x}_v$.

A pure rotation homography is trivially always a planar motion, since there is no translation component. A plane-induced homography however can exist for any motion. The interpretation of $\mathbf{x}_v$ in this case is that it is the image of the point at which the screw axis pierces the scene plane. One important point is that a plane-induced homography has a fixed line and a fixed point regardless of the motion. If the motion is not planar, $\mathbf{x}_v$ has the same interpretation (*ie.* the image of the intersection of the plane and screw axis), but the fixed line is no longer the image of the invariant line at infinity.[5]

2.5. *Alignment Using Planar Image Transformations*

With the necessary tools for decomposing 2D transforms into the relevant invariants, the algorithm for alignment remains very simple:

Algorithm 2. Single Camera Alignment Algorithm

To align axis A with respect to axis B, first make a motion about axis B, and obtain before and after images.

Calculate either a fundamental matrix $\mathtt{F}$ or 2D homography $\mathtt{H}$ relating the two images, using the method of choice.

Calculate the image of the invariant line, $\mathbf{l}_h$. For $\mathtt{H}$ it is the real eigenvector of $\mathtt{H}^{-\top}$. For $\mathtt{F}$, it is calculated from the eigendecomposition of the symmetric part of $\mathtt{F}$, as described in Section 2.3.

Fixate a point on the line. Further iterations may be required if the motion was large.

Calculating the image transformations was carried out in our implementation using what is now a fairly standard feature-based sequence (Hartley and Zisserman, 2000): detection of Harris corners (Harris and Stephens, 1988), robust two-view matching (Torr and Zisserman, 2000), followed by non-linear optimisation of the closed-form algebraic solution using Levenberg-Marquardt iteration. Any method could be used, but a non-linear minimisation stage, while not essential, has been exploited in this work not just to improve results, but to measure accuracy, correct lens distortion (Section 3.2), and allow the incorporation of information from additional images (Section 4.1).

In the case of aligning a camera on Yorick, it is simpler to make a motion about the pan axis and calculate one invariant line, then about the elevation axis and calculate the other. Then fixate the intersection of the two lines by rotating the elevation and vergence axes only, which aligns the head in a single go.

Which of `F` or `H` applies depends both on the scene and the motion. In the *general* case, where there is a general scene and a general planar motion, the fundamental matrix applies (the *F-method*). If the rotation axis passes through the camera centre (or sufficiently close), as is generally the case for an active head, or the scene is planar, then a homography applies (the *H-method*). This is the *degenerate* case. The fundamental matrix still exists (it is just underconstrained), and experiments will show that the `F`-method works equally well in either case, but is still outperformed by the `H`-method under most circumstances. This turns out to be true even when the `H`-method is used in the general case, since a homography can still be calculated (it just doesn't fit the data as accurately).

3. Results and Analysis

3.1. Simulations

Thorough testing was carried out on both the general and degenerate case algorithms. Figure 10 shows the results of these tests.

The simulated camera had a focal length of 760 pixels, an aspect ratio of 1, and a principal point in the centre of the image, which was size 640×480 pixels — similar to the digital cameras used for real scene tests. Variations on these parameters had little effect on the quality of results, so they remained fixed. The scene was populated with random points arranged in a cuboid of random orientation. The cuboid was placed 3 metres away from the camera down the optic axis, with dimensions of 4 metres in width and height, but variable depth (relief) (Fig. 9). The camera was then rotated about an axis of random orientation placed at a fixed distance from the camera centre. Next the scene was projected into the images at both camera positions, and a fixed number of point matches extracted following the addition of Gaussian noise.

There were a number of variables in the tests, including standard deviation of the feature location uncertainty (referred to as 'image noise'),[6] radial distortion, and depth (or relief) of scene. While one value was being varied, the others were fixed to standard values, except where otherwise marked on the graphs. Table 1 shows these standard values and the ranges of the variables.

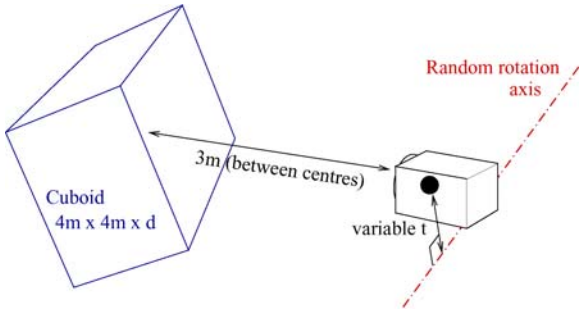


Figure 9. The scene for single camera alignment simulations consists of a random points in a cuboid volume filling the whole view.

Radial distortion was modelled up to second order, as is generally sufficient in vision applications (Tordoff and Murray, 2000). The model used gives the distorted points $\mathbf{x}_\mathrm{d}$ and undistorted points $\mathbf{x}_\mathrm{u}$, in a coordinate system whose origin is the centre of the image, as

$$\mathbf{x}_\mathrm{d} = \frac{\mathbf{x}_\mathrm{u}}{\sqrt{1 - 2\frac{\kappa\|\mathbf{x}_\mathrm{u}\|^2}{f^2}}} \qquad \mathbf{x}_\mathrm{u} = \frac{\mathbf{x}_\mathrm{d}}{\sqrt{1 + 2\frac{\kappa\|\mathbf{x}_\mathrm{d}\|^2}{f^2}}}. \quad (5)$$

The division by the square of the focal length f (in pixels) is included so that the distortion parameter κ becomes a dimensionless parameter independent of the choice of image resolution. Typical values of κ are in the region of -0.1 for a commercial camera with unspecialised optics, or a pixel motion at the edges of the image of about 4% of the image width.

For all our simulations, the y-axis shows a mean error (in statistics, the 'mean deviation'). This was chosen over RMS error (standard deviation) because it has a more direct relation to the expected accuracy of the algorithm. However both are indicators of spread. For the record, the error plotted closely tracked the median, and the distribution could be characterised as approximately Gaussian.

The simulation enforces somewhat harsh conditions, since the axis of rotation is completely random, and the axis being aligned is also a completely random vector perpendicular to that axis. Despite this the results, shown in Fig. 10, are promising. Under typical conditions, where we might typically expect to obtain more than 300 matches and an image noise of around 1 pixel, an error of around half a degree can be expected. The

Table 1. Variable settings and ranges in the simulation tests of the single camera alignment routines.

Variable	Standard value	Minimum	Maximum
Image noise standard deviation (pixels)	1.0	0.0	3.0
Angle of rotation (degrees)	10.0	1.0	30.0
Number of matched points	200	50	400
Radial distortion parameter κ	0	−0.2	0.2
Depth of scene cuboid (metres)	4	0	4
Distance between camera and axis (metres)	0.1	0.0	1.0

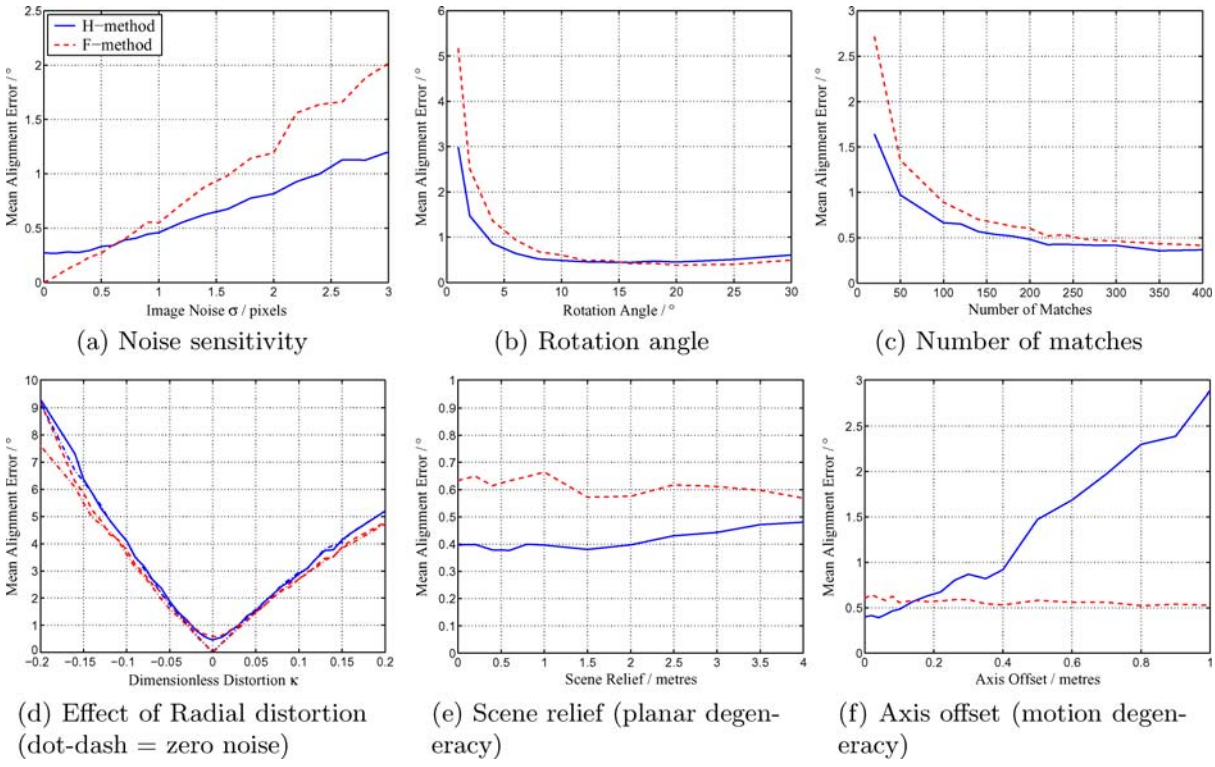


Figure 10. Results for the tests on simulated data. The blue (solid) line is alignment error when using the fixed line calculated from a homography (H-method); the red (dashed) line is that from use of the fundamental matrix (F-method). The error measure is mean absolute deviation. Default values of variables can be found in Table 1. The number of iterations made to obtain each point was 1000.

graphs also show what is to be expected, that small numbers of matches and a small rotation will impair the results. Note however that Fig. 10(b) shows that much larger a rotation than about 20 degrees may in fact not be beneficial, since the benefits of improved motion constraints are offset by the reduced overlap between images which means matched points only cover a small part of the image, leading to poorer conditioning. This can be expected to be emphasised in real scenes where reduced overlap means fewer matches as well.

Also of interest is the apparent lack of effect of axis offset for realistic values of the offset. It might be expected that a non-negligible axis offset would be precisely when the (exact) F-method would outperform the (approximate) H-method, and indeed the F-method is better for values of offset greater than 20 cm (see Fig. 10(f)). However a more typical value would be 10 cm (as it is in our robotic head, Yorick), and in this instance in the presence of typical noise (1 pixel) the H-method is clearly superior. Note, though, that although degenerate for an offset of zero or a planar scene (zero scene relief), the F-method does *not* fail catastrophically as might be expected from the fact that the epipolar geometry cannot be calculated uniquely. While some of the parameters of the matrix become poorly conditioned (at the very least the

image of the screw axis $\mathbf{l}_s$ will not be determined correctly), the invariant line parameter $\mathbf{l}_h$ remains stable, demonstrated by the continued good quality of alignment at low offsets and low scene relief (Figs. 10(e) and 10(f)).

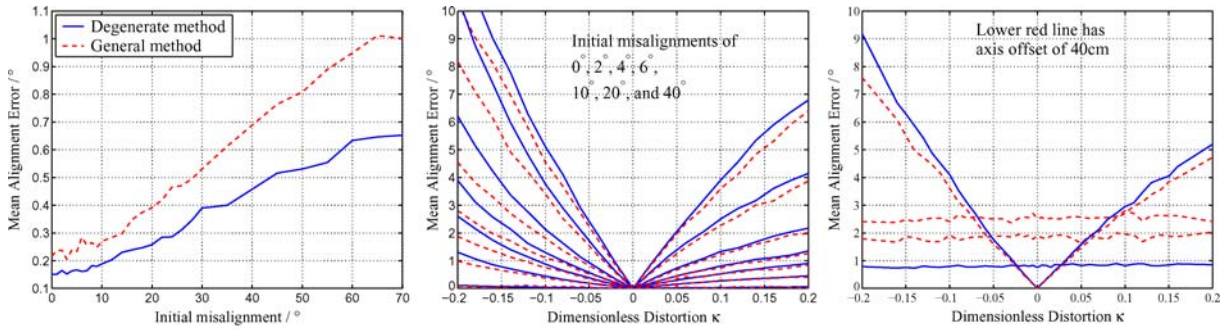
Despite this, and the fact that it is only the F-method that is theoretically capable of achieving a zero alignment error (as shown by Fig. 10(a)), the graphs suggest that unless the axis offset is unusually high, the H-method will always produce better results. This is undoubtedly because when the motion is near-degenerate a homography is better conditioned than a fundamental matrix.

3.2. *Effect of Radial Distortion and Initial Misalignment*

Probably the biggest problem appears to be the sensitivity to radial distortion. While a camera with specialised optics can expect to have a low distortion parameter which places the alignment error in the region around 1 degree, a more ordinary camera, such as the digital cameras used in this work, would appear to push the error up nearer 5 degrees. In fact Fig. 10(d) shows that distortion is the dominant factor in inducing error, since the difference between zero noise and noise of 1 pixel is negligible for all but the lowest values of κ.

Tordoff and Murray (2000) shows how barreling distortion will have the effect of causing the focal length to be over-estimated, with it potentially becoming infinite for high distortions, and for it to be underestimated under pin-cushion distortion. Distortion will therefore have similar effects to zooming. Negative distortion is like zooming in (paradoxically, since in fact it brings points closer to the distortion centre), so the invariant line will be pushed away from the centre of the image; under positive distortion the opposite will occur.[7] This is underlined by the graph of Fig. 11(b), which shows that distortion causes incorrect alignment even when there is no image noise and the matches are therefore perfect. The good news, however, is that this means the distortion should have no effect when the camera is already aligned.

The effect of initial misalignment is examined in Fig. 11. It shows that the calculation does indeed become more poorly conditioned as the initial misalignment increases and the invariant line therefore moves further away from the centre of the image, eventually going off the image completely. The second graph, Fig. 11(b), then shows that the effect of distortion is, as expected, dependent on the initial misalignment, becoming essentially zero if the head happens already to be aligned correctly. The graphs indicate that even if a lower quality camera is used, the head alignment will generally always be improved at each iteration of the algorithm, meaning that in theory the head will eventually find the true alignment, regardless of the distortion. Despite this it would be better to correct for distortion if possible.



(a) Effect of initial misalignment

(b) How initial misalignment affects the error caused by distortion (zero noise)

(c) Effect of κ correction. V-shape = no correction

Figure 11. Additional results for single camera alignment, showing the combined effects of initial alignment error and radial distortion. Once again, the solid blue line shows results for the H-method, the dashed red line for the F-method. In graph (a) the image noise is 1 pixel, while in the others it is zero. In each case the axis offset has been set to zero for the H-method and 10 cm for the F-method, to ensure that both algorithms reach zero error for zero distortion.

Since κ is dependent on the camera lenses, it is quite a stable parameter, so a value obtained from a prior calibration procedure can be expected to be usable for the lifetime of a camera.[8] Alternatively, κ can be corrected at the non-linear minimisation stage of the calculation of the fundamental matrix or homography, by the inclusion of a single additional state parameter. Figure 11(c) shows this with a comparison of the effect of radial distortion on alignment when ignoring κ.

κ was initialised at zero for self-correction, and in the case of the `H`-method successfully removes the effect of distortion. For the `F`-method the dependence on distortion is negated at the expense of an increased mean error. However, the variance of the results was large, and unless the axis offset was high there tended to be a high proportion of excessively poor alignments resulting from wildly incorrect estimation of the distortion. The conclusion is that when estimating a fundamental matrix from just two images there simply are not sufficient constraints to obtain a reliable value for the radial distortion factor.

3.3. Results in Real Scenes

Simulation provides the thorough testing that no single piece of hardware or small set of viewing conditions can achieve. However, obviously simulation circumvents the problems of occlusion affecting the distribution of point data, and various other considerations.

The single camera algorithms were tested in the real scene of Fig. 12, a fairly ordinary office environment. This was chosen simply because it contains a range of types of structure over a wide field of view. During these automated tests the vergence and elevation axes were those being aligned and their initial orientations were randomised to an extent dependent on the experiment being carried out. The pan axis, as the first axis in the kinematic chain, could be reoriented between tests without affecting the calculations; consequently it was adjusted at random over a wide angle so that the content of the scene being viewed was variable (as Fig. 12 shows, from the viewpoint of the head, which is at the bottom centre of the image, the scene varies in content and relief over a good 60°–90° of azimuth). Both algorithms were tested on planar scenes as well and it was evident that the `F`-method continued to work under those circumstances as expected.

As well as the two algorithms, the tests were carried out with a range of initial head positions, and a range of rotation angles. The methods were also tested with no correction for radial distortion, with detected points corrected for distortion using an approximate value for κ obtained from a camera calibration, and with the algorithms attempting to correct the distortion themselves at the minimisation stage. The camera used in these experiments had quite a large radial distortion (a motion at the corner of the image of about 4% of the image width). The results are shown in Figs. 13 and 14. Each point represents the RMS error (adjusted for sample bias) from fifty measurements, which is not statistically significant (hence the raggedness of the plots), but is sufficient to get a reasonable idea of the quality of results[9].

Figure 12. Left: a view of the scene used for testing the algorithm. Right: an example of three images used to obtain the fixation point, with the invariant lines, and (bottom right) the image after alignment.

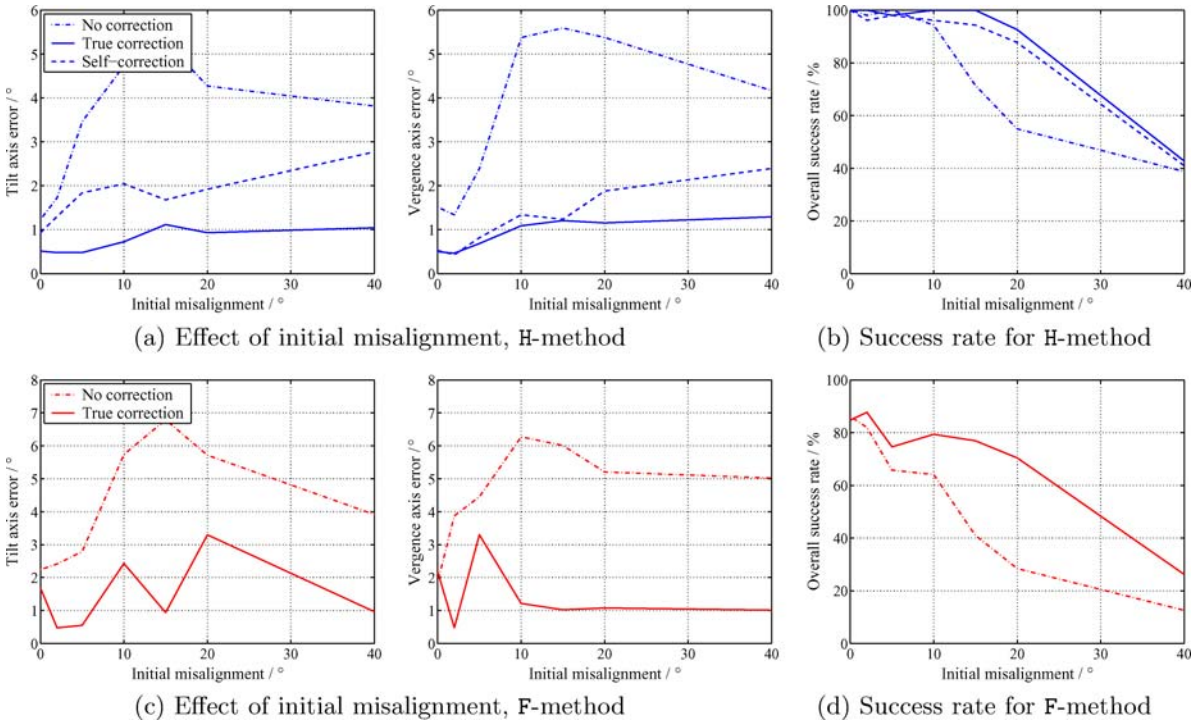


Figure 13. Results of real scene tests of the alignment algorithm, showing the effect of initial misalignment and level of correction of radial distortion. The error measure here is RMS error from best case results (ground truth being unavailable). Self-correction of distortion for the F-method was not robust enough to be viable, so those results are not shown.

In the case of variable initial head misalignment, the graph abscissa is the magnitude of that angle in the vergence and elevation axes (i.e. $\sqrt{\theta^2 + \phi^2}$), while the direction of the initial position was random. While this cannot be compared directly to the simulations, it bears a similar relationship to relative position of the invariant line and the image centre. During those tests the rotation angle used was 10°. When varying the rotation angle the initial misalignment on each axis was a random value between $\pm 10^\circ$.

In the absence of a robust and accurate method of obtaining ground truth data, the 'true' aligned position was taken from the very best results available during the real scene tests, that is, the mean alignment position for an initial misalignment of approximately zero, a rotation of 10°, and with radial distortion correctly accounted for. Slight inaccuracy, or drift, in this value may account for the error at zero initial misalignment being slightly higher than that at 2°.

If the fixation point lay off the image, the algorithm would repeat up to a maximum of 4 iterations in total. The graphs show that correcting radial distortion is quite crucial to ensuring successful and accurate alignment in a reasonable number of iterations. The worse the initial alignment, the worse the effect of distortion. If distortion is not corrected, a bad initial alignment will result in such an erroneous invariant line that the system may well get stuck in a loop of continuously overshooting the correct alignment. This was a common cause of eventual failure, the rate of which is shown in Figs. 13(b) and 13(d): either the maximum number of algorithm iterations is reached, or the alignment will be so poor that the cameras will be facing a texture-sparse scene like the ceiling or floor, and will not obtain sufficient matches to continue. Correcting distortion removes this problem, and the success rate consequently increases dramatically. Remaining failures usually occurred when the randomised initial alignment resulted in the images containing too little texture, or when the minimisation routine's residual error failed to achieve a maximum threshold.

In summary, when the best algorithm is used we can expect to be able to achieve alignments to within half a degree, as long as radial distortion is small or corrected, and as long as the process is allowed to iterate until further alignments no longer move the head significantly.

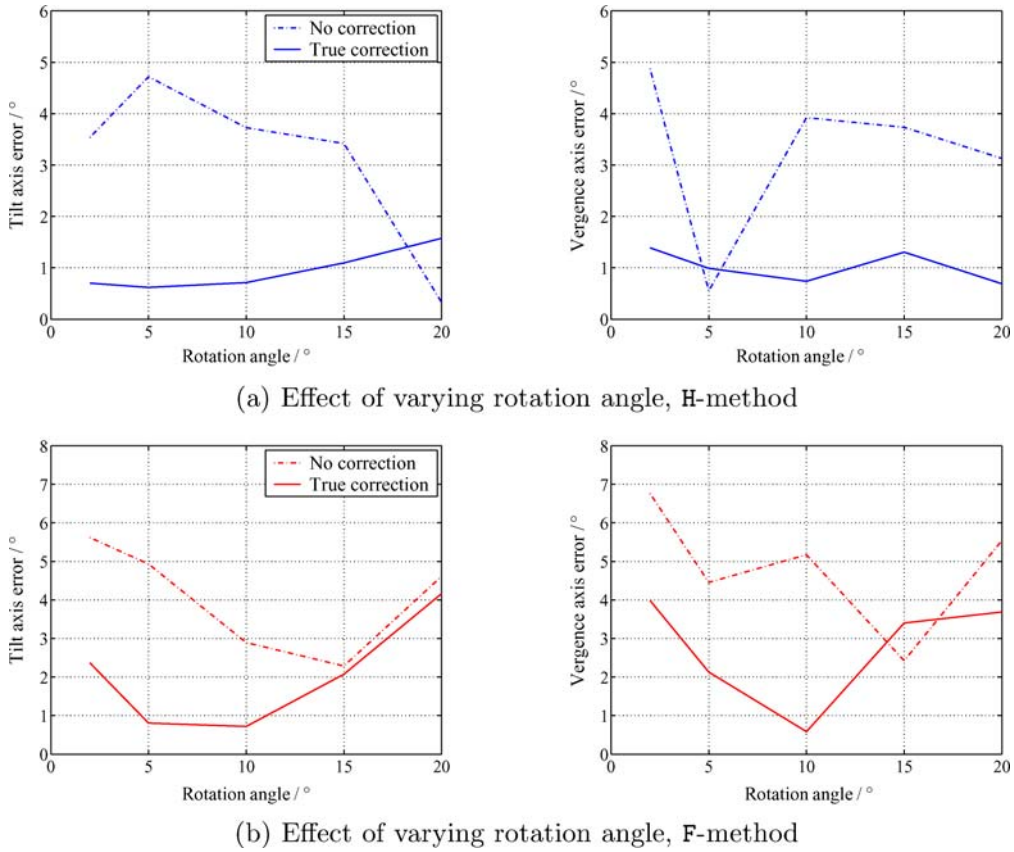


Figure 14. Real scene results showing dependence of the monocular algorithm on the angle of rotation.

The H-method still out-performs the F-method, despite the variation in the depth of scene structure. This, combined with the failure of self-correction of distortion when estimating the fundamental matrix, leads again to the conclusion that the H-method should be used unless there is a good reason to use the fundamental matrix algorithm.

Figure 14 suggests that the alignment will sometimes be improved by using an increased rotation. However, higher rotation angles also experienced an increased failure rate due to insufficient matches being obtained. The best rotation angles would still seem to be around 10° to 15° for these cameras, or 20–30% of the image width in general.

3.4. Conclusions for Alignment from Pairs of Images

In conclusion, an implementation of alignment should, as a set of general rules: use the H-method; correct for radial distortion where it is significant; iterate as long as the misalignment continues to be greater than 5°–10°; reject calculations where the number of matches is too low or the residual error from the minimisation routine is too high; and align 'off' images containing significant texture.[10] In addition, if it is possible to provide an initial alignment by eye then doing so will reduce the number of iterations required to align the head; alternatively a rough alignment can be achieved using direct methods: axis A is aligned with respect to axis B when the rotational component of optic flow is minimised during motion about B.[11]

Sources of Error. Our tests show that radial distortion is the greatest source of error, hence the need to correct for it. However, once this is done we are still limited by the digital and optical resolution of the camera, and by the nature of the scene, all of which affect our ability to localise invariant features for matching.

Figure 10(a) tells us that where this matching noise is the only source of error, at around 1 pixel standard

deviation we can expect about 0.5° error: compare the real scene tests in Fig. 13(a), which show a similar error for a roughly pre-aligned head. It is possible, then, that the system was exhibiting around 1 pixel of error, just about that expected from digitisation alone. Consequently, one route to improving accuracy would be to use a higher resolution digital camera.[12]

Note again that these results are for two views per aligned axis only. Improved accuracy can be obtained by combining image features from multiple motions of the camera, as is now discussed.

4. Extensions to Incorporate Further Motions

Perhaps the simplest way to use additional images and motions is to take the mean alignment position over multiple independent tests.

Figure 15 demonstrates the viability of this technique. In Fig. 15(a) we see that continued iteration does improve alignment even in the presence of radial distortion, as expected. Figure 15(b) shows that once the major initial misalignment is removed, averaging does result in convergence to the true alignment. The graphs show the 95th percentile alignment error, which removes some unusually high errors that crop up during simulation due to unrealistic configurations, but otherwise effectively represents a *guaranteed* level of accuracy after the relevant number of iterations. The exception is the F-method which cannot provide these guarantees unless there is high positive distortion (the resulting undershoot providing a stabilising influence). Despite this the median error for the F-method is, like the H-method, around 0.1°.

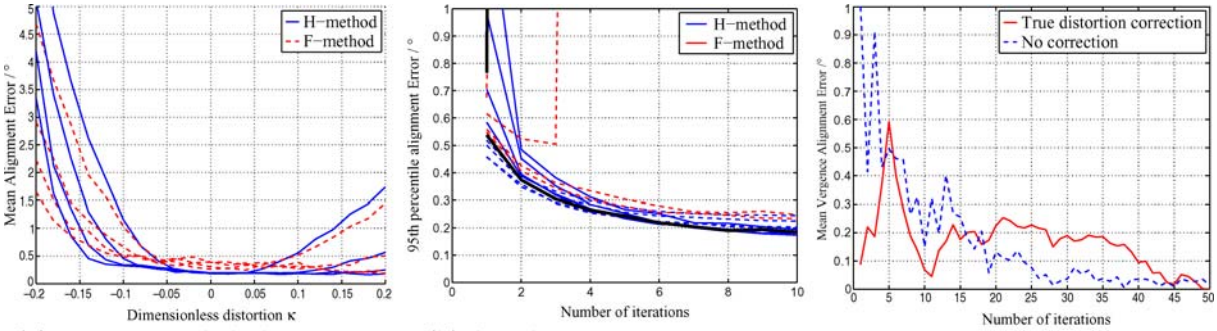
In Fig. 15(c) we then see how this translates into real scenes, with a slow convergence over many iterations. We also confirmed that the distribution of alignment positions was unbiased even in the presence of distortion, and that the mean and mode of this distribution lay in the same place whether or not the distortion was corrected. Radial distortion does of course produce bias (towards or away from the true alignment), but over multiple iterations this is negated because noise eliminates dependence on the starting orientation.

While averaging is a statistically sound solution when carried out over a large number of iterations, it is somewhat *ad hoc*, and places equal weight on each iteration regardless of the quality of the solution it provides. For each motion we recalculate the F/H matrix.

However each motion of the head is identical, except for one free parameter (the angle of rotation). Therefore it makes intuitive sense to exploit this by parameterising the motions (i.e.the F/H matrices) explicitly to encode this fact. We explore this below.

4.1. *Batch Minimisation*

Our aim is to parameterise the motion in terms of the fixed entities. In the camera coordinate frame, the only parameter of the motion that is varying is the angle of



(a) Varying radial distortion for 2 to 5 motions

(b) Simulation: Worst or near worst case errors after averaging

(c) Real tests: Convergence for the H-method only

Figure 15. Convergence when averaging over multiple independent alignments. In (a), simulation results for different levels of radial distortion demonstrating that iterating will always improve the alignment. In (b), simulation results from multiple tests of 10 iterations following removal of the initial bias. Each line of the same colour represents a different level of radial distortion, varying evenly between ±0.2. The solid lines are for the negative distortions and the dashed lines for the positive distortions. The thick black line is for zero distortion (this line for the F-method is mostly off-scale). (If graphs are being viewed in grayscale, it suffices to note that convergence is good regardless of distortion, except for the F-method and negative distortion.) In (c), convergence in real scenes from a single long test.

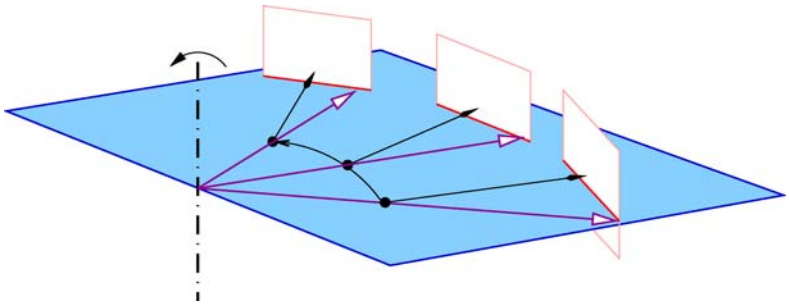


Figure 16. Illustrating how multiple planar motions about the same axis result in images with the same invariant line (i.e.in the same image position). Note also that the screw axis projects to the same line (or point) in each image.

rotation. In our case the (horizon) line $\mathbf{l}_h$ is invariant. This quantity is precisely what we seek for alignment, and therefore our aim in this section is to parameterise $\mathtt{F}$ and $\mathtt{H}$ in terms of $\mathbf{l}_h$. This can then be shared between views and some or all of the others allowed to vary. By doing so we expect to obtain a better conditioned problem.

Multiple Image General Case Alignment. In Section 2.3 two parameterisations of a planar fundamental matrix are described. The Vieville and Lingrand parameterisation (Viéville and Lingrand, 1996) includes $\mathbf{l}_s$ and $\mathbf{l}_h$, which is desirable, but also involves trigonometric functions and the somewhat obscure θ parameter; while Hartley and Zisserman (2000) suggest using $\mathbf{l}_s$ and the epipoles. We compromise by parameterising the epipoles in terms of $\mathbf{l}_s$ and the fixed line $\mathbf{l}_h$ (the epipoles lie on $\mathbf{l}_h$), so that $\mathbf{l}_h$ becomes one of the parameters of the simpler Hartley and Zisserman form.

To do this, first note that any linear combination of two points $\mathbf{x}_1$ and $\mathbf{x}_2$ must lie on the line between the two points, $\mathbf{l} = \mathbf{x}_1 \times \mathbf{x}_2$:

$$\begin{aligned} \mathbf{x}_3 &= \alpha\mathbf{x}_1 + \beta\mathbf{x}_2 \\ \Rightarrow \mathbf{l}^\top\mathbf{x}_3 &= \alpha\mathbf{l}^\top\mathbf{x}_1 + \beta\mathbf{l}^\top\mathbf{x}_2 \\ \Rightarrow \mathbf{l}^\top\mathbf{x}_3 &= 0 \end{aligned}$$

This also means that any point $\mathbf{x}_3$ on $\mathbf{l}$ can be parameterised in terms of $\mathbf{x}_1$ and $\mathbf{x}_2$, even if $\mathbf{x}_3$, or one of $\mathbf{x}_1$ or $\mathbf{x}_2$, is at infinity.

So to parameterise the epipoles $\mathbf{e}$ and $\mathbf{e}'$ in terms of the fixed lines, obtain the intersection of $\mathbf{l}_s$ and $\mathbf{l}_h$, $\mathbf{x}_s = \mathbf{l}_s \times \mathbf{l}_h$. Next obtain another point on $\mathbf{l}_h$ perpendicular to that, $\mathbf{x}_p = \mathbf{x}_s \times \mathbf{l}_h$ (see Fig. 17). Now find the parameters α, β, α' and β' from the following, ensuring $\mathbf{x}_s$ and $\mathbf{x}_p$ are normalised so that $\mathbf{x}_{s|p}^\top \mathbf{x}_{s|p} = 1$:

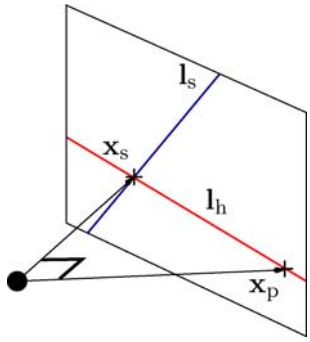


Figure 17. When parameterising the epipoles in terms of the lines $\mathbf{l}_s$ and $\mathbf{l}_h$, use their intersection $\mathbf{x}_s$, and a point on $\mathbf{l}_h$ perpendicular to this, $\mathbf{x}_p$. As long as the two lines, which can be seen as representing vectors perpendicular to the planes that pass through them and the camera centre, have consistent directions (which can be ensured in the minimisation routine), then $\mathbf{x}_p$ will always be calculated on the same side of $\mathbf{x}_s$.

$$\begin{aligned} \mathbf{e} &= \alpha\mathbf{x}_s + \beta\mathbf{x}_p & \mathbf{e}' &= \alpha'\mathbf{x}_s + \beta'\mathbf{x}_p \\ \Rightarrow \alpha &= \mathbf{x}_s^\top\mathbf{e}, \quad \beta = \mathbf{x}_p^\top\mathbf{e} & \alpha' &= \mathbf{x}_s^\top\mathbf{e}', \quad \beta' = \mathbf{x}_p^\top\mathbf{e} \end{aligned}$$

The resulting parameters for the fundamental matrix are $\mathbf{l}_s$, $\mathbf{l}_h$, α, β, α' and β', and $\mathtt{F}$ can be obtained from these as

$$\begin{aligned} \mathtt{F} = (\alpha'[\mathbf{l}_s]_\times\mathbf{l}_h + \beta'[\mathbf{l}_s]_\times[\mathbf{l}_h]_\times\mathbf{l}_h) \times \mathbf{l}_s \times (\alpha[\mathbf{l}_s]_\times\mathbf{l}_h \\ + \beta[\mathbf{l}_s]_\times[\mathbf{l}_h]_\times\mathbf{l}_h). \end{aligned}$$

$\mathbf{l}_h$ is to be shared between all the motions about the same axis. $\mathbf{l}_s$ could also be shared since the screw axis has not moved. However it was found empirically to be insufficiently well constrained, so better results were obtained by allowing it to vary.

Multiple Image Degenerate Case Alignment. The simplest way to parameterise a sequence of planar motion homographies is to share the eigenvectors while allowing the eigenvalues to alter. The eigenvectors represent the fixed line and fixed point, both of which will be unchanged during a motion about the same axis. The eigenvalues represent the scale of the matrix (which is irrelevant and will be ignored by Levenberg-Marquardt) and the angle of rotation.

A homography will have two complex eigenvalues and eigenvectors, but they will be conjugate, so it is only necessary to store the real and imaginary parts of one. The complex eigenvalue is further constrained. If the real eigenvalue (representing the scale) is λ_1 and the complex eigenvalues are λ_2 and λ_3, then $\lambda_2 = \lambda_1 e^{\imath\theta}$ and $\lambda_3 = \lambda_1 e^{-\imath\theta}$, where θ is the angle of rotation. If this constraint is not expressed the homography may become biased, so it is preferable to record only λ_1 and θ as unshared parameters.[13]

Since in general the homography is only an approximation to a pure rotation (due to the offset of the rotation axis), there is no guarantee that any of the eigenvectors are genuinely fixed. Despite this, tests showed that the fixed entities do not move significantly between motions, and there are distinct advantages to sharing all the eigenvectors between motions for any realistic axis offset value.

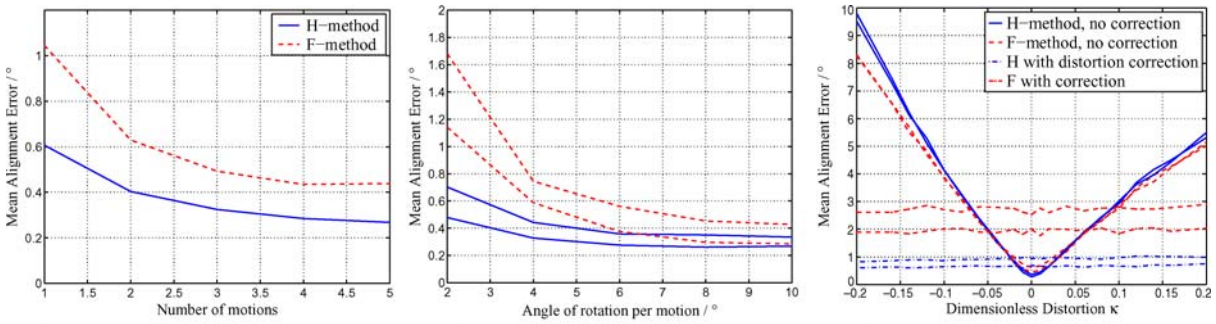
Multiple Motion Implementation and Results. Our implementation involves the system making several motions about the same axis, and obtaining the F/H matrices for each rotation. One of the calculated transforms is then used to find an initial guess for the invariant line, which is the starting point for a bundle adjustment over the line parameters and remaining free parameters.

In practice, Yorick has a limited vertical field of view, particularly indoors (due to the sparsity of scene features above and below) which might limit the accuracy of vertical alignment. However, when making elevation motions the pan axis can be rotated towards different parts of the scene without affecting the position of the invariant line (due to the order of the kinematic chain).

The set-up for tests of the multiple motion algorithms on synthetic data was identical to that for a single motion. Variables were the number of motions, the angle of rotation for each motion, and the radial distortion. Generally the axis offset was a randomised value between zero and 0.4 m, a realistic maximum range. However for the radial distortion experiment the axis offset was set once again to 0.1 m, for comparison with the single motion tests.

The results are much as expected, with additional motions improving alignment. The H-method has a minimum error related to the axis offset. However it still outperforms the F-method, which itself does not appear to be tending towards a perfect result in the limit. In fact the graphs of Fig. 18(a) are consistent with those for varying number of point matches of Fig. 10(c), suggesting that whether additional matches are obtained in a single pair or from multiple pairs of images, noise places a limit on the achievable accuracy[14].

Figure 18(b) shows that good results can be obtained even with small rotations. This is advantageous, since in a real scene the increased overlap for a small rotation provides a greater number of matches, as well as better conditioning.



(a) Effect of additional 5° rotations

(b) Varying angle of rotation for 2 and 4 motions

(c) Effect of κ with or without correction, for 2 and 4 motions

Figure 18. Results for the tests of monocular multiple motion alignment with synthetic data. Results for the H-method are shown in solid blue, with dashed red for the F-method. Dot-dashed lines show the effect of radial distortion correction using an initial value of zero.

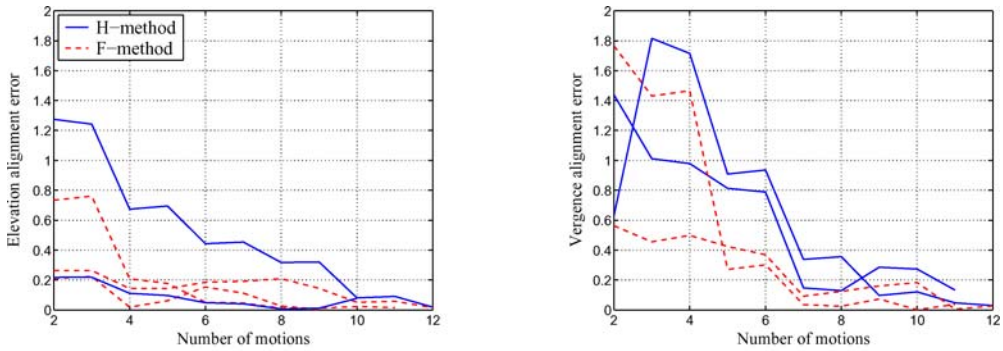
It is interesting to note that multiple motions do not help with the problem of radial distortion (Fig. 18(c)), however self-correction of distortion is improved. Self-correction is still poor for the F-method, although over large numbers of moves it could be expected to become accurate. However, once again, in the general case for a typical pan-tilt unit, there appears to be little advantage in the F-method, despite its theoretical greater accuracy in the limit.

Real tests were carried out on the same scene as before (Fig. 12). The camera was rotated by 5° alternately about the pan and elevation axes. Several tests were done at different initial positions, and some typical examples are shown in Fig. 19.

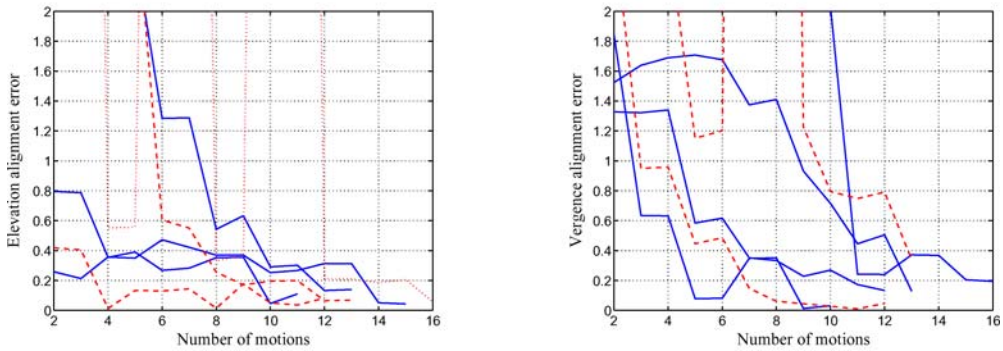
As expected, the tests showed a general improvement with the inclusion of additional images, and an improvement in the radial distortion detected using self-correction. In addition, this method reduces the reliance on making a large rotation for each motion, and on obtaining large numbers of matches in each pair of images. It has advantages over simple averaging of accuracy as well as a more meaningful weighting of the data from each motion.

An Aside on Dual Model Minimisation. An interesting point to note is that the invariant line $\mathbf{l}_h$ is a parameter shared by both the homography and fundamental matrix, and could possibly be used to provide additional constraints on both. There are many scenarios in which both types of transformation can be calculated, but neither are well constrained; this is particularly true of active heads, which make near-degenerate motions, and of near-planar scenes.

In this dual model minimisation the fundamental matrix would be parameterised as before in terms of $\mathbf{l}_h$, $\mathbf{l}_s$, and the epipoles; the homography would be parameterised using the eigendecomposition of $\mathtt{H}^{\top}$, the inverse line-to-line mapping, so that $\mathbf{l}_h$ is the real eigenvector.



(a) True radial distortion correction



(b) Self-correction of distortion

Figure 19. Some results for different real scene test runs using multiple images, showing the general trend towards an improved alignment, regardless of the initial accuracy from a single motion. The dotted red line is an example of an F-method alignment for which the minimisation occasionally failed to converge to a good value for the distortion.

If we use reprojection error or its approximation the Sampson error (Sampson, 1982) for our cost functions, then the costs of H and F can be summed because they both represent image distances between measured points and their 'true' positions.

5. Summary and Conclusions

In order to be able to use an active camera as a measuring device it must be possible to obtain pointing directions referred to a fixed coordinate frame. Any mechanical method of specifying the origin of this frame suffers from being hardware-specific, to the extent that the camera itself must have a strongly fixed mounting on its robotic platform. It also requires precision engineering, which counters the trend towards using cheaper products and compensating for hardware deficiencies in software. Image-based methods get around these problems, but most of the solutions involve carrying out a complete kinematic calibration of the device (Li, 1998; DeSouza et al., 2002; Ma, 1996; McLauchlan and Murray, 1996; Hayman et al., 2000). This is generally overkill, with a complex implementation and requiring lengthy processing.

In this paper we have presented an image-based solution which solves only for the relevant alignment parameters from a bare minimum of input images, doing so using a fast and robust algorithm that places no constraints on the viewing conditions, other than the presence of sufficient texture for image correspondence to be feasible. The system has been tested thoroughly, and we have demonstrated how to handle the problem of non-linear lens distortion, and how to improve accuracy by incorporating the information from additional images.

Notes

1. The qualification "that control gaze direction" is used to exclude cyclorotation which, if controllable, has a related natural origin defined by the other axes, therefore evading a completely generalised definition. The cyclotorsional orientation of the camera is defined by its image plane (here, the x-axis); cyclorotation is zero when this axis is perpendicular to a vertical head axis (pan); or its normal (the y-axis if there is no skew) is perpendicular to a horizontal head axis (elevation). Alignment of a cyclotorsion axis is not tested in this paper, but our algorithm is perfectly capable of achieving this, and Section 2 addresses the necessary implementation.
2. In this paper 'pan' and 'verge(nce)' refer to the vertical head axes, i.e. those controlling azimuthal gaze, with pan being the principal azimuthal axis (and usually first in the kinematic chain), and vergence generally only being present for binocular arrangements. 'Elevation' and 'tilt' are used interchangeably to mean a horizontal axis controlling the elevation of the gaze direction.
3. The problem of fixating an image point without first knowing the head and camera calibrations can be addressed using simple discrete closed-loop control. The gain of the controller is adjusted at each iteration by using correlation to measure the image distance moved under the previous input (Knight and Reid, 2000a; Knight, 2002).
4. It should be noted that no failures were detected in this matching algorithm which were distinguishable from general failures due to insufficient constraints on F.
5. The fixed line in this case has no simple geometric representation. In all cases it is the line between the two intersection points of the *horopter curve* (Maybank, 1993) and the scene plane. In the general case (non-planar motion including translation) this line is dependent on the amount of rotation as well as the position and orientation of the rotation axis. A special case for which the invariant line remains as the horizon line of the plane of the motion was exploited for camera calibration in Knight et al. (2003).
6. Image noise here is more correctly viewed as the combination of noise in the sensor, and the view-dependent bias inherent in feature detection.
7. We should therefore expect the alignment error to asymptote for some value of large negative distortion, but tend to the initial misalignment for high positive distortion.
8. κ does of course change with zoom, so it would need to be calibrated at each zoom position that might be used when aligning the head.
9. RMS error is used here, as opposed to mean deviation, because it provides a smoother indication of spread for limited sample size. Since we cannot obtain ground truth other than by using the best results available, we are restricted to providing a measure of spread to indicate repeatability.
10. If scene texture is sparse it may be possible to calculate optical flow but not a sufficient number of localised features: this gives further advantage to the H-method, since a homography can be estimated using optic flow.
11. This is the same as minimising the cyclotorsion component of camera motion, as in the method of Hayman et al. (2000).
12. This would also improve the accuracy of the fixation algorithm, localisation of image features in each case being interrelated.
13. If the homographies are divided through by λ_1 then it may be discarded and assumed to be 1 in each case, but a minimal parameterisation is not essential, neither is it advisable because it complicates the error surface of the cost function (Hartley and Zisserman, 2000).
14. This limit is around 0.2° for our setup, but would be improved with a higher resolution camera as suggested in Section 3.4.

References

Armstrong, M.N. 1996. Self-calibration from Image Sequences. Ph.D. thesis, University of Oxford.

Beardsley, P.A. and Zisserman, A. 1995. Affine Calibration of Mobile Vehicles. In R. Mohr and W. Chengke (Eds.): *Proc. Joint Europe-China Workshop on Geometrical Modelling and Invariants for Computer Vision*, Xi'an, China.

Davison, A. 1998. Mobile robot navigation using active vision. Ph.D. thesis, Robotics Research Group, Oxford University Department of Engineering Science.

DeSouza, G.N., Jones, A.J., and Kak, A.C. 2002. A world independent approach for the calibration of mobile robotics active stereo heads. In *Proc. IEEE Int'l Conf. on Robotics and Automation*, vol. 4. Washington, pp. 3336–3341.

Harris, C.G. and Stephens, M. 1988. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conf.*, Manchester. pp. 147–151.

Hartley, R.I. and Zisserman, A. 2000. *Multiple View Geometry in Computer Vision*. Cambridge University Press.

Hayman, E., Knight, J., and Murray, D.W. 2000. Self-alignment of an active head from observations of rotation matrices. In *Proc. 15th IEEE Int'l Conf. on Pattern Recognition*, vol. 1. Barcelona, pp. 80–84.

Horaud, R. and Csurka, G. 1998. Self-calibration and euclidean reconstruction using motions of a stereo rig. In *Proc. 6th Int'l Conf. on Computer Vision*, Bombay. pp. 96–103.

Knight, J. 2002. Towards fully autonomous mobile robot navigation. Ph.D. thesis, Robotics Research Group, Oxford University Department of Engineering Science.

Knight, J. and Reid, I. 2000a. Active visual alignment of a mobile stereo camera platform. In *Proc. IEEE Int'l Conf. on Robotics and Automation,* Vol.4. San Francisco, pp. 3203–3208.

Knight, J. and Reid, I. 2000b. Binocular self-alignment and calibration from planar scenes. In Vernon, D. Ed.): *Proc. 6th European Conference on Computer Vision*, Dublin. pp. II:462–476.

Knight, J., Zisserman, A., and Reid, I. 2003. Linear auto-calibration for ground plane motion. In *Proc. IEEE Conf. on Computer vision and Pattern Recognition*, Madison, Wisconsin.18–20 June, pp. I:503–510.

Li, M. 1998. Kinematic calibration of an active head-eye system. *IEEE Transactions on Robotics and Automation* 14(1):153–157.

Ma, S.D. 1996. A self-calibration technique for active vision systems. *IEEE Trans. Robotics and Automation* 12(1):114–120.

Maybank, S.J. 1993. *Theory of Reconstruction from Image Motion*. Springer-Verlag Berlin:

McLauchlan, P.F. and Murray, D.W. 1996. Active camera calibration for a head-eye platform using the variable state-dimension filter. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18(1):15–21.

Murray, D.W., Reid, I.D., and Davison, A.J. 1996. Steering and navigation behaviours using fixation. In *Proc. 7th British Machine Vision Conf.*, pp. Edinburgh. 635–644.

Murray, D.W., Reid, I.D., and Davison, A.J. 1997. Steering without representation using active fixation. *Perception* 26(12):1519–1528.

Reid, I.D. and Beardsley, P.A. 1996. Self-alignment of a binocular head. *Image and Vision Computing* 14(8):635–640.

Sampson, P.D. 1982. Fitting conic sections to 'very scattered' data: An iterative refinement of the Bookstein algorithm. *Computer Vision, Graphics, and Image Processing* 18:97–108.

Sharkey, P.M., Murray, D.W., Vandevelde, S., Reid, I.D., and McLauchlan, P.F. 1993. A modular head/eye platform for real-time reactive vision. *Mechatronics* 3(4):517–535.

Tordoff, B. and Murray, D.W. 2000. Violating rotating camera geometry: The effect of radial distortion on self-calibration. In *Proc. 15th IEEE Int'l Conf. on Pattern Recognition*, vol. 1. Barcelona, pp. 423–427.

Torr, P.H.S. and Zisserman, A. 2000. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding* 78(1), 138–156.

Viéville, T. and Lingrand, D. 1996. Using singular displacements for uncalibrated monocular vision systems. In *Proc. 4th European Conf. on Computer Vision, Cambridge*. pp. II:207–216.

Zisserman, A., Beardsley, P.A., and Reid, I.D. 1995. Metric Calibration of a stereo rig. In *Proc. IEEE Workshop on Representations of Visual Scenes, in conjunction with ICCV'95*, Cambridge, MA. pp. 93–100.