Epipolar Geometry for Humanoid Robotic Heads

Justin Hart, Brian Scassellati, and Steven W. Zucker

Department of Computer Science
Yale University
New Haven, Connecticut 06520, USA
justin.hart@yale.edu, scaz@cs.yale.edu, steven.zucker@yale.edu

Abstract. Stereoscopic vision is a capability that supports the ability of robots to interact with visually complex environments. Epipolar geometry captures the projective relationship between the cameras in a stereo vision system, assisting in the reconstruction of three-dimensional information. However, a basic problem arises for robots with active vision systems whose cameras move with respect to each other: the epipolar geometry changes with this motion. Such problems are especially noticeable in work with humanoid robots, whose cameras move in order to emulate human gaze behavior. We develop an epipolar kinematic model that solves this problem by building a kinematic model based on the optical properties of a stereo vision system. We show how such a model can be used in order to update the epipolar geometry for the head of a humanoid robot.

1 Introduction

While stereo vision provides one of the richest feedback pathways for inferring such structure from our physical environment, to utilize advanced stereo computer vision techniques that are most relevant to biological perception [1] [2] requires knowledge of the imaging system's epipolar geometry. However the world rarely stands still, and on platforms where the cameras can move independently of one other the epipolar geometry will change with this motion. Here we develop epipolar kinematic models, or kinematic models that track the motion of optical properties of the system. The result is that motor data is used to compute an updated representation of the epipolar geometry. Particular emphasis in this paper is placed on computing such models for humanoid robotic heads.

Camera calibration is the process of measuring the parameters necessary for quantitative interaction with the 3D Euclidean world. The intrinsic parameters, which include focal length, principal point, and a skew factor relating the x and y axes, describe the camera itself. The extrinsic parameters, position and orientation, describe its pose in space. Additionally, lens distortion is often modeled. It has been a heavily researched topic in the computer vision and photogrammetry communities. [3] and [4] both provide excellent overviews of prior work and are seminal papers on the topic.

B. Caputo and M. Vincze (Eds.): ICVW 2008, LNCS 5329, pp. 24–36, 2008.

In a stereo vision system, epipolar geometry describes the projective relationship between two camera views, and can either be computed from their calibration [5], or estimated for uncalibrated cameras via methods such as the 8-point algorithm [6]. In the case of calibrated cameras, the epipolar geometry is described by the essential matrix. In the case of uncalibrated cameras, it is referred to as the funamental matrix.

An active vision system is a vision system in which either the cameras are able to move or they are attached to a device that is able to manipulate its environment. Such systems include cameras mounted on robotic arms, often referred to as *hand cameras*, and also in the heads of humanoid robots, such as our upper-torso humanoid infant, Nico, which is discussed in more depth in Section 5.1. The desire to calibrate the position of the cameras relative to the underlying robotic platform has given rise to two tasks, hand-eye and head-eye calibration, which either describe solving for how a camera is mounted with respect to a movable platform, usually with known kinematics, or solving for its position in space with respect to a manipulator, [7][8][9][10][11]. Kinematic calibration is the process of estimating the kinematics of the underlying system, [12][13].

Moving cameras present a unique challenge to robotics and vision researchers who wish to exploit the epipolar geometry of multiple cameras to perform stereo vision tasks. Such a scenario arises whenever a humanoid robot performs a saccade, a tracking motion, or when the eyes verge upon an attended to object. In this paper, we discuss the relationship between camera calibration, the estimation of epipolar geometry, and the kinematics of active vision systems. Prior work on this problem has focused on the use of 3D data to estimate ego-motion visually [14], tracking points in a stereo pair [15], or developing kinematic models by detaching the cameras from the head and viewing it using an external vision system [12][13].

The central contribution of this paper is the notion of an epipolar kinematic model, which is a kinematic model based on the motion of optical properties of the projective relationship between the cameras in a stereo active vision system as the cameras move through space. From this model we can compute current epipolar geometry using only knowledge of the current angles of the motors. To demonstrate, we will build such a model for our upper-torso humanoid. This model will be suitable for use with many. We present results from a preliminary implementation of the algorithm.

2 Background

2.1 The Pinhole Camera Model

Following standard notation, as found in [5], let X denote the homogeneous representation of a point in 3-space, and x its image. When discussing a stereo pair of cameras, determine one of the two cameras to be the *first* camera. All properties of the second camera will be marked with a $'$. For instance, let x

represent the image of X in the first camera and x' the image in the second camera.

The camera projection matrix, Equation 1, represents the projection of a 3D point, X, by a camera to a 2D point, x.

$$x = PX \tag{1}$$

Modeling the camera under the standard pinhole camera model, the camera calibration matrix, Equation 2 captures the camera's intrinsic parameters, which are properties of the camera itself. α and β, express focal length and are generally equal, and γ, which is the skew factor between the x and y axes, is generally 0. u_0 and v_0 represent the principal point. Together, they define the camera calibration matrix

$$A = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

The extrinsic parameters, R, the rotation of the camera, and C, the camera center are combined with the camera calibration matrix as in Equation 3 to yield the camera projection matrix. These parameters can be retrieved via a number of standard camera calibration methods[4][3].

$$P = A[R \mid -RC] \tag{3}$$

2.2 Epipolar Geometry

Under the pinhole camera model, image points are represented as rays of light intersecting the *image plane* on a line running through the camera center. Given a pair of cameras, P and P', and a point x in camera P, we can constrain the position of x', the image of the same three-dimensional point X in P' to a line, l'. The image of one camera's camera center in the other camera is called an epipole. This system is called epipolar geometry, because these epipolar lines must all run through the epipole.

This relationship can be captured by the *fundamental matrix*, F, Equation 4.

$$x'^T F x = 0 \tag{4}$$

Given calibrated cameras, we can express our points as normalized image coordinates, coordinates corresponding to the same camera, but with A equal to the identity matrix. We express our coordinate system with in terms of P, giving us $P = [I|0]$ and $P' = [R| - RC]$. In this case, our essential matrix can be expressed as in Equation 5. The relationship between E and F is Equation 6. The fundamental and essential matrices can be computed using standard techniques [6][5][16][17].

$$E = [-RC]_\times R \tag{5}$$

$$F = A'^{-T} E A^{-1} \tag{6}$$

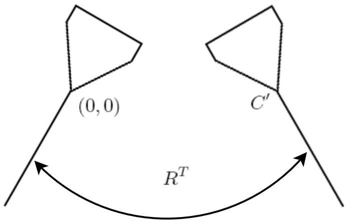


Fig. 1. Setup of the extrinsic parameters in the epipolar geometry problem. We define our coordinate system with the first camera at the origin. The second camera is rotated by R^T.

3 Epipolar Kinematics

In the case of stereo vision systems in which the cameras remain stationary with respect to each other, it is enough to estimate the epipolar geometry once, via the familiar process of matching control points in each image to each other and computing the projective relationship between them for 8 or more points [5]. For vision systems such as Nico's, however, this estimate will become inaccurate the first time that the robot moves its eyes.

Approaches have been demonstrated that cope with this via purely visual means, such as optical flow [15][14]. While a number of stereo tasks can still be performed via algorithms that do not require knowledge of epipolar geometry, such approaches ignore kinematic information available to the system that can be used to maintain the estimate in "real time." Most significantly, there is evidence that primate visual processing is structured precisely to take advantage of this [18].

From the formulation in Section 2.1, we can update the essential matrix with respect to camera motion provided that we know the way that the cameras move with respect to each other. Note that here we specifically mean the motion of the camera center and orientation of our pinhole cameras, optical properties that can be retrieved via standard computer vision techniques. One of the central insights of this work is that we can estimate our kinematic models based on these optical properties. This allows us to build our models using only image data processed by our stereo vision system with its joints turned in several orientations, rather than requiring for us to preprogram the system's kinematics or externally calibrate the kinematics of our visual system [12][13].

4 Epipolar Kinematics for Humanoid

We define this model as a kinematic model over two revolute joints. This is reflective of those degrees of freedom relevant to the epipolar geometry of the head of our our humanoid robot, Nico, as well as those of many other humanoid robots.

Backlash is not modeled, and should be considered on top of this model if it is a significant concern. Finally, assume that the camera faces directly away from the axis of rotation of the joint on which it is mounted. We can easily eliminate this assumption, but retain it because it reduces the number of measurements that must be made, resulting in a faster calibration process, and also because it accurately describes the vision systems on most humanoid robots. We feel that the community of researchers working with humanoid robots is the most likely group to incorporate this method into their work. In Section 7, we will briefly discuss how to eliminate this assumption as well as how to model more complicated kinematic systems.

Our epipolar kinematic calibration algorithm is agnostic to the methods used for camera calibration and estimation of epipolar geometry. As such, we present this as a framework into which preferred methods for these two processes can be plugged in. By turning the linkage on which the camera is mounted and observing the relationship of this view to the view before turning the camera we can deduce the kinematics of the system. If that system has the constraint that the camera faces directly away from the center of rotation, as it does on Nico, then we are able to uncover the kinematics of that linkage by observing as few as two views.

4.1 Calibration Algorithm

Initial measurement. Proceed by choosing two angles for each of the eye motors to be calibrated. Denote the first camera in the first orientation, $Cam_{1,1}$, in the second orientation, $Cam_{1,2}$, the second camera in the first orientation

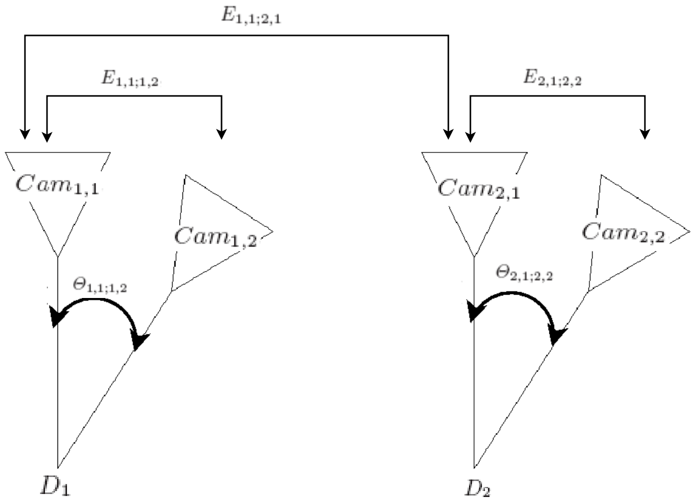


Fig. 2. Camera orientations and variables used in this process

$Cam_{2,1}$, and so forth for all parameters of the system, as in Figure 2. Let $E_{1,1;1,2}$ be the essential matrix between the $Cam_{1,1}$ and $Cam_{1,2}$, $E_{2,1;1,2}$ between $Cam_{1,1}$ and $Cam_{2,1}$ and so forth. As is standard, in the below discussion we will treat the first orientation in each essential matrix having its rotation matrix equal to the identity matrix and its camera center at the origin.

Calibrate the cameras and solve for $E_{1,1;1,2}$, $E_{2,1;2,2}$, and $E_{1,1;2,1}$.[1] Use any preferred method for both processes.[2]

Kinematic rotation axis and angle. Let $R_{1,1;1,2}$ be the rotation matrix found by decomposing $E_{1,1;1,2}$. If $V_{1,1;1,2}$ is the corresponding rotation vector, found by Rodrigues' Rotation Formula [19], then $\Theta_{1,1;1,2}$ is the magnitude of $V_{1,1;1,2}$, Equation 7.

$$\Theta = ||V|| \tag{7}$$

Dividing by Θ yields a unit axis of rotation, S, Equation 8.

$$S = \frac{V}{\Theta} = \frac{V}{||V||} \tag{8}$$

Kinematic link length. The camera centers for a camera before and after motion, such as $C_{1,1}$ and $C_{1,2}$, and the center of rotation of the epipolar kinematic linkage form an isosceles triangle. Therefore, the length of the linkage is given by Equation 9.

$$L_1 = \frac{\frac{||C_{1,2}||}{2}}{sin(\frac{\Theta_{1,1;1,2}}{2})} \tag{9}$$

Finding the center of rotation. Per our assumption that the camera faces directly away from the axis of rotation, compute the center of rotation via Equation 10, where D_1 is the center of rotation for for the first camera. D_2 is computed analogously.

$$D_1 = C_{1,1} - [0\ 0\ L_1] \tag{10}$$

Algorithm. Our entire calibration algorithm is summarized in Algorithm 1. This algorithm need be performed only once, at startup. The update rule to estimate the updated essential matrix is presented in Section 4.2.

[1] The purpose of the essential matrices is to ground the coordinate system with respect to the first camera orientation in each. If the calibration method places all of the extrinsic parameters in the same coordinate system, this step can be ignored, and the epipolar kinematic calibration process modified accordingly.

[2] The reader who is familiar with these processes will note that this might yield multiple values for A for the same physical camera, hopefully remarkably close to one another. In our implementation, we calibrate the cameras once using OpenCV.

Algorithm 1. Estimation of system parameters

```
1. Calibrate the two cameras in the stereo active vision system, $Cam_1$ and $Cam_2$
2. Choose 4 camera orientations, two for each camera, $Cam_{1,1}$, $Cam_{1,2}$, $Cam_{2,1}$,
   $Cam_{2,2}$
3. Estimate the essential matrix in each orientation
4. Factor $E_{1,1;1,2}$, $E_{2,1;2,2}$, $E_{1,1;2,1}$ per Equation 5
5. Compute $V_{1,1;1,2}$ and $V_{2,1;2,2}$ from $R_{1,1;1,2}$, $R_{2,1;2,2}$ via Rodrigues' Rotation Formula
6. Compute $\Theta_{1,1;1,2}$ and $\Theta_{2,1;2,2}$, per Equation 7
7. Compute $S_1$ and $S_2$ per Equation 8
8. Compute $L_1$ and $L_2$ per Equation 9
9. Compute $D_1$, $D_2$ per Equation 10
```

4.2 Updating the Essential Matrix

At runtime, we update our essential matrix to reflect the new position and orientation of the cameras each time they move with respect to each other. This means updating every time the motors move, changing this relationship.

Let $\Theta_{1,Enc}$ be the difference between the Θ indicated by the encoder at $Cam_{1,1}$ and the current encoder reading for that camera's associated motor. Let $\Theta_{2,Enc}$ be the analogous value for the second camera. All variables subscripted Enc will be with respect to the current encoder reading. Compute updated $V_{1,Enc}$, $V_{2,Enc}$ by Equation 11.

$$V_{1,Enc} = \Theta_{1,Enc} * S_1 \tag{11}$$

Update $R_{1,Enc}$ and $R_{2,Enc}$ via Rodrigues' Rotation Formula. We'll denote variables reflective of the current position and rotation of the second camera with respect to the current position and rotation of the first camera by subscripting them $CurSys$. Let R_{CurSys} is given by Equation 12.

$$R_{CurSys} = R^T_{1,Enc} * R_{2,Enc} * R_{1,1;2,1} \tag{12}$$

Find the updated camera centers, $C_{1,Enc}$ and $C_{2,Enc}$ via Equation 13.

$$C_{1,Enc} = R^T_{1,Enc}[0\ 0\ L_1] - D_1 \tag{13}$$

Find the updated camera center in the second view with respect to the first view, C_{CurSys}, Equation 14. Remember that $C_{1,Enc}$ and $C_{2,Enc}$ do not share the same world coordinate system.

$$C_{CurSys} = C_{2,1} - C_{1,Enc} + C_{2,Enc} \tag{14}$$

Compute the updated essential matrix, Equation 15.

$$E_{CurSys} = A_2'^{-T}[-R_{CurSys} * C_{CurSys}]_\times R_{CurSys} A_1^{-1} \tag{15}$$

Our entire update algorithm is summarized in Algorithm 2. Since this algorithm involves only constant-time matrix computations, it can be used to update the epipolar geometry in real-time.

Algorithm 2. Essential matrix update

```
1. Compute $V_{1,Enc}$, $V_{2,Enc}$ per Equation 11. Compute updated $R_{1,Enc}$, $R_{2,Enc}$ from $V_{1,Enc}$, $V_{2,Enc}$ via Rodrigues' Rotation Formula
2. Compute $R_{CurSys}$ via Equation 12
3. Compute $C_{1,Enc}$, $C_{2,Enc}$ via Equation 13
4. Compute $C_{CurSys}$ via Equation 14
5. Compute $E_{CurSys}$ via Equation 15
```

5 Tests

5.1 Platform

Nico, Figure 3, is an upper-torso humanoid robot that has been modeled after the the kinematic structure of a fiftieth percentile 12-month-old male infant. It has 23 mechanical degrees of freedom, including seven in each arm and two in its recently-added hand. Its head has six degrees of freedom and employs a foveated vision system consisting of four NTSC color cameras mounted in two anthropomorphic eyes. The eyes have mechanically coupled pitch and independent yaw degrees of freedom. Nico's compute platform includes a 20-node cluster running the QNX real-time operating system. Nodes are connected via 100 Mbit Ethernet to each other and to a number of Linux and Windows machines whose configurations change from experiment to experiment. This architecture allows

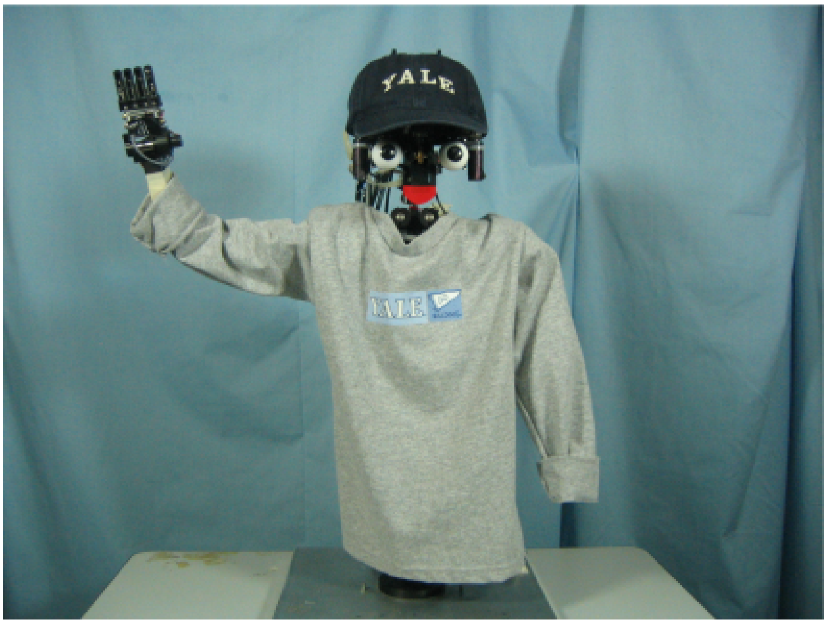


Fig. 3. Nico, an upper-torso humanoid infant

us to easily integrate software packages on multiple platforms into Nico's control system, as well as to remotely operate Nico via the Internet.

5.2 Test Setup

In order to test our system we took 3 sets of images of chessboards. Imaging proceeded as follows. The cameras were first to -10 degrees, then to 0 degrees, 5 degrees, and 10 degrees. A chessboard was placed in front of the robot and it was visually confirmed that the robot could locate all of the interior corners of the chessboard in both cameras in the last 3 of these orientations. The first orientation was then returned to in order to assure that any backlash was worked out of the system. It is expected that this, combined with the fact that the system uses zero-backlash motors worked out most of the backlash. Several images were taken in each position for each orientation of the chessboard in order to generate a set of images for both camera calibration and the estimation of epipolar geometry.

Using these images, the cameras were calibrated using OpenCV [20], and the essential matrices between the 0 and 10 degree views for each motor, and the 0 degree views for both motors were computed using a third-party tool, *Gandalf*.[3]

Tests were performed on the images shot at 5 degrees. For comparison, we computed the essential matrix both using the epipolar kinematics algorithms and directly from the images.

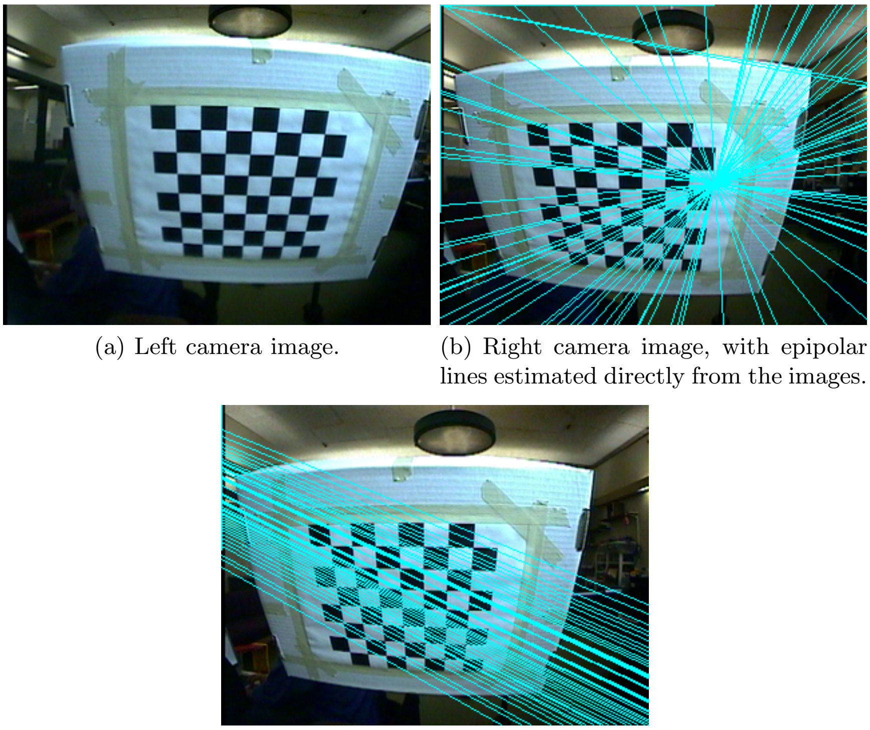
6 Results

The results included in this section should be regarded as preliminary, as the accuracy in our essential matrix estimates does not match the sub-pixel resolution expected from state of the art algorithms. Updated results will be made available in a future publication.

Upon testing, we found that the software package we used to estimate the essential matrix, Gandalf, exhibits a degree of numerical instability that is quite common in software that is used to estimate epipolar geometry [21]. In order to work around this instability, we built a software package that processed all possible subsets of image pairs for each essential matrix to Gandalf. As an error metric, we adopted the mean distance in pixels between an epipolar line and its corresponding point. We computed the epipolar lines in the right image corresponding to chessboard corners in the left image and measured the distance to the corresponding image point. We chose each essential matrix as the one corresponding to the lowest mean distance for each matrix required to compute the epipolar kinematic model, as well as for the essential matrix computed directly from the test images. See Table 2 for the mean distance corresponding to each matrix.

The redundancy of the Θ yielded by both checking the encoder readings and the computation of the essential matrix gives us the opportunity to check our robot's physical readings against those estimated by the vision algorithm. Results of this comparison are listed in Table 1. As we can see, there is significant

[3] http://gandalf-library.sourceforge.net/



(a) Left camera image.

(b) Right camera image, with epipolar lines estimated directly from the images.

(c) Right camera image, with epipolar lines estimated using the essential matrix update algorithm.

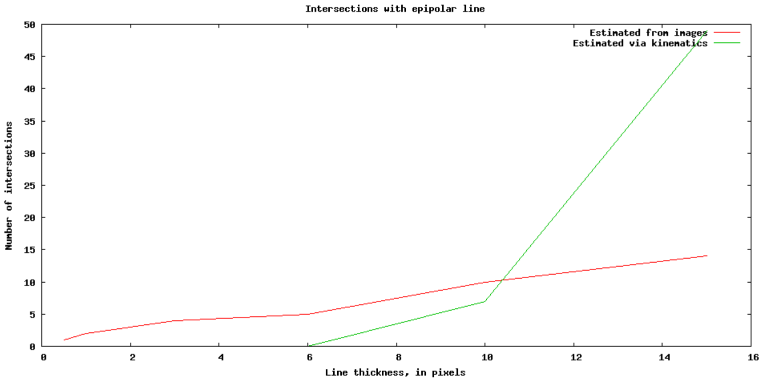


Fig. 4. Number of intersections from epipolar line to corresponding point plotted against line thickness in pixels. There are 49 test points.

disagreement between the vision algorithm and the motors. Potential sources of this error include backlash in the motors and gears, and error in the estimates of the essential matrix or camera calibration.

Table 1. Θ's estimated by essential matrix computation

	Θ_a	Θ_b	Θ_c
Estimated from E	0.17412	0.0479997	0.0109251
Turned to	-	10	10

Table 2. Mean distance from a given epipolar line to its corresponding point

Essential Matrix	Mean distance from point to epipolar line
Input to the epipolar kinematic model	
$E_{1,1;1,2}$	1.11736
$E_{2,1;2,2}$	0.638143
$E_{1,1;2,1}$	2.38185
Results	
Directly Computed From Images	18.7897
Epipolar Kinematic Model	11.0749

The epipolar lines yielded by these computations appear in Figures 4(a), 4(b), and 4(c). As we can see, the positioning of the epipole in the two images is not the same. As a test of the relative quality of the two algorithms, we took the mean distance from the epipolar line computed from the chessboard corners in the first image to the corresponding point in the second image using both an essential matrix estimated directly from imaged points and one estimated using an epipolar kinematic model. Results can be seen in Table 2. Another rough estimate of the quality of the essential matrix is the number of epipolar lines that intersect with their corresponding image points. We compare the two matrices in Figure 4.

7 Conclusion

The primary insight offered in this paper is that we can build epipolar kinematic systems, kinematic systems built directly off of the optical properties of a stereo vision system, in order to track these properties as the system moves. This allows us to keep a consistent view of the epipolar geometry of the system as it undergoes motion.

To demonstrate this technique, we showed how to compute the epipolar kinematics of the degrees of freedom of the active vision head on our humanoid robot, Nico, for those degrees of freedom effecting the system's epipolar geometry. The algorithms in this paper are suitable for the active vision heads of many humanoid robots. In this exploration we can clearly see that the estimation of epipolar kinematics is built on top of the existing suite of techniques available to the tasks of camera calibration and estimation of epipolar geometry.

Though a version of this algorithm that uses two orientations per camera is presented in this paper, it is possible to update this algorithm to use three orientations per camera in order to lift the assumption that the cameras face straight forward from center of rotation. In this case, we are able to estimate the circle defining the rotation from the three camera centers, found during the estimation of epipolar geometry, per camera, leaving us only to solve for the rotation of the camera with respect to the endpoint of the linkage. Such an algorithm is equivalent to solving head/hand-eye calibration and kinematic calibration simultaneously. Building on this process, we can estimate the epipolar kinematics of systems where more than one linkage can control the orientation of the camera. The same mathematics can equivalently be used to solve for the kinematics of a manipulator, or other kinematic linkage visible in the visual field, and its relationship to the coordinate system of the visual system. This is all deferred to future work in which the robot learns about its self in terms of its kinematics and its sensors.

Acknowledgements

Support for this work was provided by a National Science Foundation CAREER award (#0238334) and NSF award #0534610 (Quantitative Measures of Social Response in Autism). This research was supported in part by a software grant from QNX Software Systems Ltd and support from the Sloan Foundation. S.W.Z. is additionally supported by AFOSR, NSA, and NSF.

References

1. Li, G., Zucker, S.W.: Contextual inference in contour-based stereo correspondence. Int. J. of Computer Vision 69(1), 59–75 (2006)
2. Li, G., Zucker, S.W.: Differential geometric consistency extends stereo to curved surfaces. In: Proceedings of the 9th European Conference on Computer Vision, pp. III: 44–57 (2006)
3. Zhang, Z.: A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(11), 1330–1334 (2000)
4. Tsai, R.Y.: A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. IEEE Journal of Robotics and Automation, 221–244 (1992)
5. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, Cambridge (2004)
6. Deriche, R., Zhang, Z., Luong, Q.T., Faugeras, O.D.: Robust recovery of the epipolar geometry for an uncalibrated stereo rig. In: Eklundh, J.-O. (ed.) ECCV 1994. LNCS, vol. 800, pp. 567–576. Springer, Heidelberg (1994)
7. Shih, S.W., Hung, Y.P., Lin, W.S.: Head/eye calibration of a binocular head by use of single calibration point. In: Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 154–159 (1994)
8. Li, M., Betsis, D.: Head-eye calibration. In: ICCV 1995: Proceedings of the Fifth International Conference on Computer Vision, Washington, DC, USA, p. 40. IEEE Computer Society, Los Alamitos (1995)

9. Tsai, R., Lenz, R.: Real time versatile robotics hand/eye calibration using 3d machine vision. In: Proceedings of IEEE International Conference on Robotics and Automation, 1998, April 24-29, 1988, vol. 1, pp. 554–561. IEEE Computer Society Press, Los Alamitos (1988)
10. Shiu, Y., Ahmad, S.: Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form ax=xb. IEEE Transactions on Robotics and Automation 5(1), 16–29 (1989)
11. Tsai, R., Lenz, R.: A new technique for fully autonomous and efficient 3d robotics hand/eye calibration. IEEE Transactions on Robotics and Automation 5(3), 345–358 (1989)
12. Shih, S.W., Jin, J.S., Wei, K.H., Hung, Y.P.: Kinematic calibration of a binocular head using stereo vision with the complete and parametrically continuous model. In: Casasent, D.P. (ed.) Proc. SPIE, Intelligent Robots and Computer Vision XI: Algorithms, Techniques, and Active Vision, November 1992. The Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, vol. 1825, pp. 643–657 (1992)
13. Shih, S.-W., Hung, Y.-P., Lin, W.-S.: Kinematic parameter identification of a binocular head using stereo measurements of single calibration point. Proceedings of the IEEE International Conference on Robotics and Automation 2, 1796–1801 (1995)
14. Bjorkman, M., Eklundh, J.-O.: A real-time system for epipolar geometry and egomotion estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 506–513 (2000)
15. Björkman, M., Eklundh, J.O.: Real-time epipolar geometry estimation of binocular stereo heads. IEEE Trans. Pattern Anal. Mach. Intell. 24(3), 425–432 (2002)
16. Luong, Q.-T., Faugeras, O.: The fundamental matrix: Theory, algorithms, and stability analysis. International Journal of Computer Vision 17(1), 43–75 (1996)
17. Hartley, R.I.: In defence of the 8-point algorithm. In: ICCV 1995: Proceedings of the Fifth International Conference on Computer Vision, Washington, DC, USA, p. 1064. IEEE Computer Society Press, Los Alamitos (1995)
18. Dobbins, A.C., Jeo, R.M., Fiser, J., Allman, J.M.: Distance modulation of neural activity in the visual cortex. Science (281), 552–555 (1998)
19. Faugeras, O.: Three-dimensional computer vision: a geometric viewpoint. MIT Press, Cambridge (1993)
20. Intel Corporation: Open Source Computer Vision Library: Reference Manual (1999-2001)
21. Izquierdo, E., Guerra, V.: Estimating the essential matrix by efficient linear techniques. IEEE Transactions on Circuits and Systems for Video Technology 13(9), 925–935 (2003)