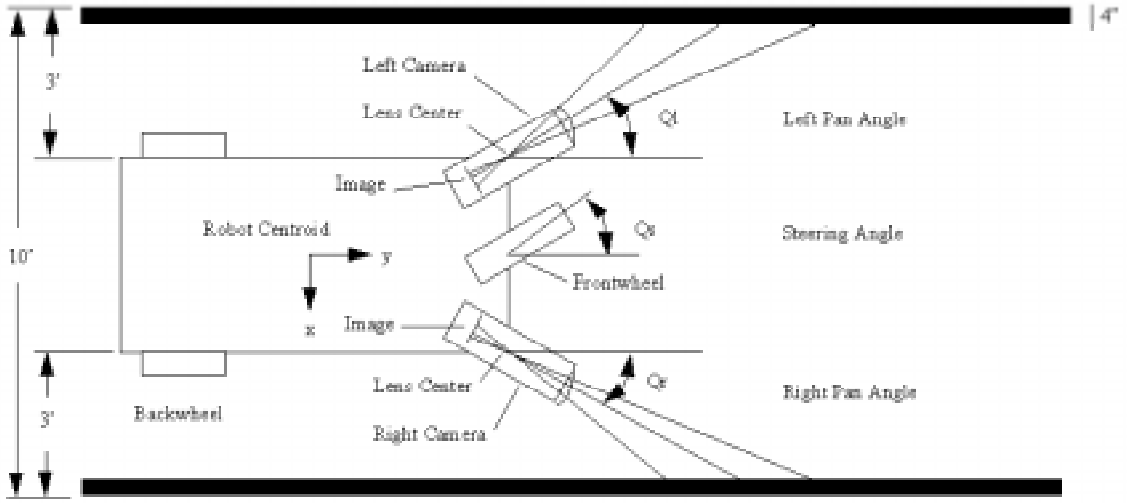
# Vision System for a line following Unmanned Autonomous Mobile Robot

**Tayib Samu**

**Center for Robotics Research**

**University of Cincinnati**

**Cincinnati, OH 45221**




## Introductic

Humans receive a large amount of their information through the human vision system as such it enables them to adapt quickly to changes in their environment. An intelligent machine such as mobile robot that must adapt to the changes of its environm ent must also be equipped with vision system so that it can collect visual information and use this information to adapt to its environment. The center for robotics research at the University of Cincinnati has been involved in a nation wide competition to building a small-unmanned autonomous ground vehicle that can navigate around an outdoor obstacle course. The major components of the vehicle are: the supervisor control computer, speed control, steering control, obstacle avoidance, breaking system, emer gency controls, and vision system. The purpose of the vision system is to obtain information from the changing environment - the obstacle course. The robot then quickly adapts to this information through its controller that guide the robot to follow the obstacle course. Modeling of the vision system is done with a CCD camera and that is the focus of this study.

The obstacle course that the robot is suppose to follow is bounded by a solid and dashed lines ten feet apart and it can assume different shapes. Via the medium of the CCD camera, the lines of the obstacle course are digitized by the vision system from a 3-D coordinate system to a 2-D coordinate system. An image-processing tool for the vision system displays

the image of the line. A 3-D coordinate system has been reduced to 2-D coordinate system by the camera system. The robot from the vision s ystem easily obtains the information about the 2-D image coordinates. In an autonomous situation, how can the 3-D information or coordinates of a line be determined given its image coordinate? A mathematical as well as geometrical transformation occurs via the camera parameters in transforming a 3-D coordinate system to a 2-D system. If these mathematical and geometrical relations are known, a 3-D coordinate point on a line can be autonomously determined from its corresponding 2-D image point. To esta blish these mathematical and geometrical relationships, the camera has to be calibrated. The calibration of the camera or the vision system is the main task of this project. This is because if the vision system is well calibrated, accurate measurements of the coordinates of the points on the line with respect to the robot will be made. From these measurements, the orientation of the line with respect to the robot can be computed. With these computations, the next task is to guide the robot and this is a control problem.

In [1], Hall discussed the fundamental theorem of robot vision. The manipulation of a point in space $x_1$ by either a robot manipulator that moves it to another point $x_2$ or through a camera system that images the point onto a ca mera sensor at $x_2$, is described by the same matrix transformation, which is of the form

$$x_2 = Tx_1$$

The transformation matrix T can describe the first-order effects of translation, rotation, scaling, and projective and perspective projections. The robot vision theorem suggests that the sensing of a point or collection of points on an object hav e some relation. In an effort to exploit this relation, the calibration of the sensing becomes very essential.

Camera calibration is a complex problem because of the following problems [2]: (1) All the intrinsic and extrinsic parameters should be computed from the two-dimensional projections of a limited number of feature points, (2) the parameters are int er-related, and (3) the formulation is non-linear due to the perspectivity of the pin-hole camera model.

Several works has been done on calibration of cameras for various applications. Lovenitti, et al [3] presented a 3-D coordinate measurement technique that uses single 2-D image of four coplanar points, which are arranged in a square of known size, to measure geometric features on an object, some of which may be hidden from the view of the camera. In this approach, a hand-held probe is positioned on the object and in the view of a camera. The perspective projection of the probe is used to determine the 3-D coordinates of the point of contact. The image of the square's four vertices is used to calculate the probe position and point of contact on the object. The method for determining the probe position is dependent on a knowledge of the probe side length, the image coordinates for each of the four vertices, and camera parameters: the image distance (distance between the sensor or the image and the optical center of the lens-the effective focal length), the lens distortion, the horizontal pixel widt h and vertical pixel height of the CCD sensor, and the location of the optical center of the lens on the image frame buffer. These camera parameters are determined by calibration. Some of the factors that can affect accuracy when determining position us ing image analysis techniques are lens distortion and horizontal pixel width.

A lot of caution is required during calibration. Hong et al [4] list two points that should be considered in camera calibration: 1. Reducing the location error of image features as far as possible, by exploiting image processing technique, 2. Compe nsating system error by the optimal pattern of approximating residual error of image points, namely the posterior compensation of the system error. Based on these two points, the calibration process discussed in [4] are of three parts: (1) The direct tran sformation error approximation camera calibration algorithm; (2) the subpixel image feature location algorithm combined with the 3D control point field delicate design and fabrication; (3) The precisely movable stage, which provides the reliable means of accuracy checking.

A method based on locating a few three-dimensional coordinates and their corresponding image coordinates on the image plane may be used to obtain the perspective projection matrix elements. The use of the corner points of a cube known dimensions a s the reference points to measure object surface is described by Parke [5].

In [6], Renner described a method that uses 23 miniature light bulbs on a pyramid base to serve as the reference points. Measurements of these reference points require the use of accurate and reliable measuring devices. In a study done by Tio [7], an object was partitioned into grids and the selection of points was based on a widely dispersed set of points that provide information on each major grid line. These reference points were used with the image points on sets of matrix equations to obtain th e parameters for the calibration of the camera.

Tsai[8] presented an algorithm that decomposes a solution for 12 transformational parameters (nine for rotation and three for translation) into multiple stages by introducing a radial alignment constraint. The radial alignment constraint assumes t hat the lens distortion occurs only in the radial direction from the optical axis Z of the camera. Using this constraint, six parameters are computed first, and the constraint of the rigid body transformation is used to compute five other parameters. Th e remaining parameters are computed by radial lens distortion parameter and estimating it by a nonlinear optimization procedure.

Liu et al. [9] first suggested the use of straight lines and points as features for estimating extrinsic camera parameters. Line features are usually abundant in indoor and some outdoor environments and less sensitive to noise than point features. The constraint used for the algorithms is that a three-dimensional line in the camera coordinate system (X, Y, Z) should lie in the plane formed by the projected two-dimensional line in the image plane and the optical center. This constraint is used fo r computing nine rotation parameters separately from three translation parameters. They present linear and nonlinear algorithms for solutions. According to Liu et al's analysis, eight-line or six-point correspondences are required for the linear method, and three-line or three-point correspondences are required for the nonlinear method. A separate linear method for translation parameters requires three-line or two point correspondences. Haralick et al. [10] reported their comprehensive investigation f or position estimation from two-dimensional and three-dimensional sensed features. Other approaches based on different formulations and solutions include Kumar [11], Yuan [12], and Chen[13].

In this study, the calibration of the vision system was done with a specially constructed

calibration device. First, three mathematical and transformational models were formulated to represent the camera parameters. For each of the models, a set of physical world coordinates and their corresponding image coordinates were obtained to solve for the unknown variables in the mathematical models. Accurate measurement of the physical points was very essential. The calibration device, which is made of a wooden base and six pin-pong balls, enabled us to achieve high accuracy. The three calibration models were evaluated and the one with the best performance was implemented on the robot.

To control the robot to follow the obstacle course, information about the angle of the line as well as the position of the robot with respect to the line is required. The vision system provides this information. Also considered in this study is a control method to guide the robot to follow the course using the information from the vision system. A look-up table with two inputs, angle of line and distance of robot from the line, and one output, the steering angle, was generated and coded in C++ t o model the control system.

Chapter 2 presented the mathematical and transformational models of the vision system with all their respective equations. The geometrical model of the robot is also illustrated in this chapter. In chapter 3, an experimental method, which describ es the construction of the calibration device, was related. The calibration process was described. Results of the calibration process for each of the calibration models were given and illustrated with graphs and tables.

Experimental results for the vision system was described in chapter 4. The calibration model with the best performance in terms of computational accuracy was implemented on the vision system to ascertain and reaffirm its accuracy. Correlation plo ts and tables were used to present the results.

The control part of the study was described in chapter 5. The computation of the inputs to the control system as well as a graphical illustration of why both inputs were required was shown in this chapter. Also presented in this chapter was the l ook-up table for modeling the control system. Chapter 6 has the conclusions and the recommendations made for this study. References are in chapter 7. The codes generated for the look-up table is given in Appendix A.

## 2.0 Mathematical Model of the Vision System

The purpose of the vision system is to guide the robot to follow a line using a digital camera. To do this, the camera needs to be calibrated. Camera calibration is a process to determine the relationship between a given 3-D coordinate system (world coordinates) and the 2-D image plane a camera perceives (image coordinates). More specifically, it is to determine the camera and lens model parameters that govern the mathematical or geometrical transformation from world coordinates to image coo rdinates based on the known 3-D control field and its image. The CCD camera digitizes the line from 3-D coordinate system to 2-D image system. Since the process is autonomous, the relationship between the 2-D system and the 3-D system has to be accurate ly determined so that the robot can be appropriately controlled to follow the line. The objective of this section is to show how three different mathematical models were developed to calibrate the vision system so that, given any 2-D image coordinate poin t, the system can mathematically compute the corresponding ground coordinate point.

### 2.1 Model of Robot Geometry

The model of the mobile robot illustrating the transformation between the image and the object is shown in Figure 1. The robot is designed to navigate between two lines that are spaced 10 feet apart. The lines are nominally 4 inches wide but are sometimes dashed. This requires a two-camera system design so that when a line is missing, the robot can look to the

other side via the second camera. Measurements are referenced to the robot centered as a global coordinate system. For navigation, the cameras must be located with respect to this centroid.

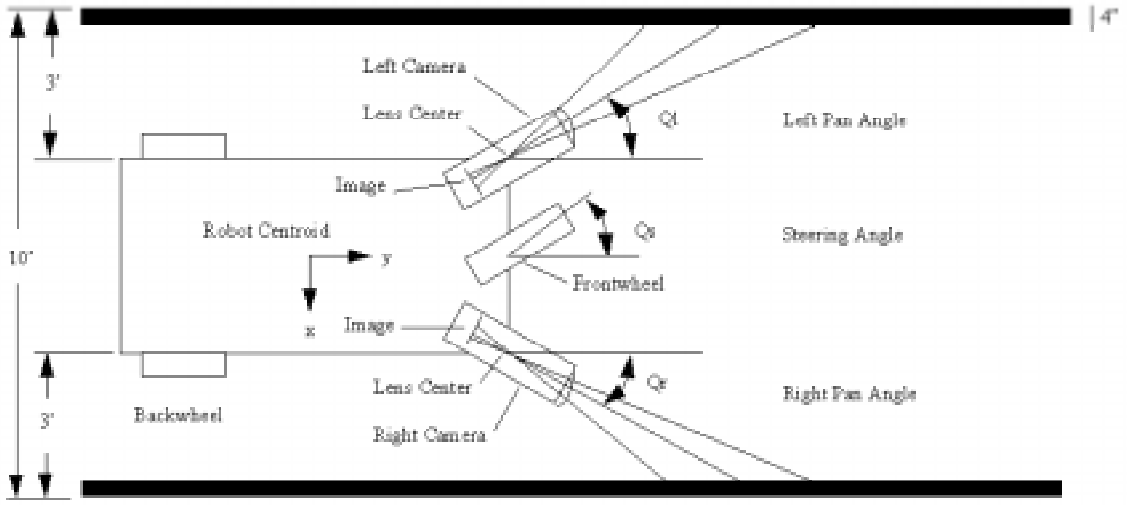



Figure 1. Top view model of the robot in the obstacle course

### 2.2 Vision Processing Devices

Point or line tracking is achieved through the medium of a digital CCD camera. An Iscan[14] tracking device does image processing. This device finds the centroid of the brightest or darkest region in a computer-controlled w indow, and returns its X, Y image coordinates as well as size information. This information is updated every 16 ms, however the program must wait 10 ms after moving the window to get new data. This results in a 52 ms update time for tracking two points in sequence [15].

### 2.3 Mathematical Models

Three mathematical models were considered for developing the vision system. The models are the forward and inverse homogeneous matrix transformation, stereo vision model approach, and direct coefficient computation approach.

### 2.3.1 The Homogeneous Matrix transformation

The homogeneous matrix transformation approach maps the physical coordinates to the image coordinates using matrix transformation. A point in global coordinates is related to the corresponding point in image coordinates using exactly the same transformations that are encountered in the physical system[3 (ref to proposal)]. The global coordinates are first translated to the center of the image plane, then rotated counter clockwise about the x-axis by the tilt angle, q , then rotated counter clockwise about the z-axis by the pan angle, $\phi$ , then mapped through a perspective transformation with lens center located at $y=y_0$, and finally projected onto the x-z plane. These series of mappings are implemented as shown below. The imaging system model is shown in Figure 2.

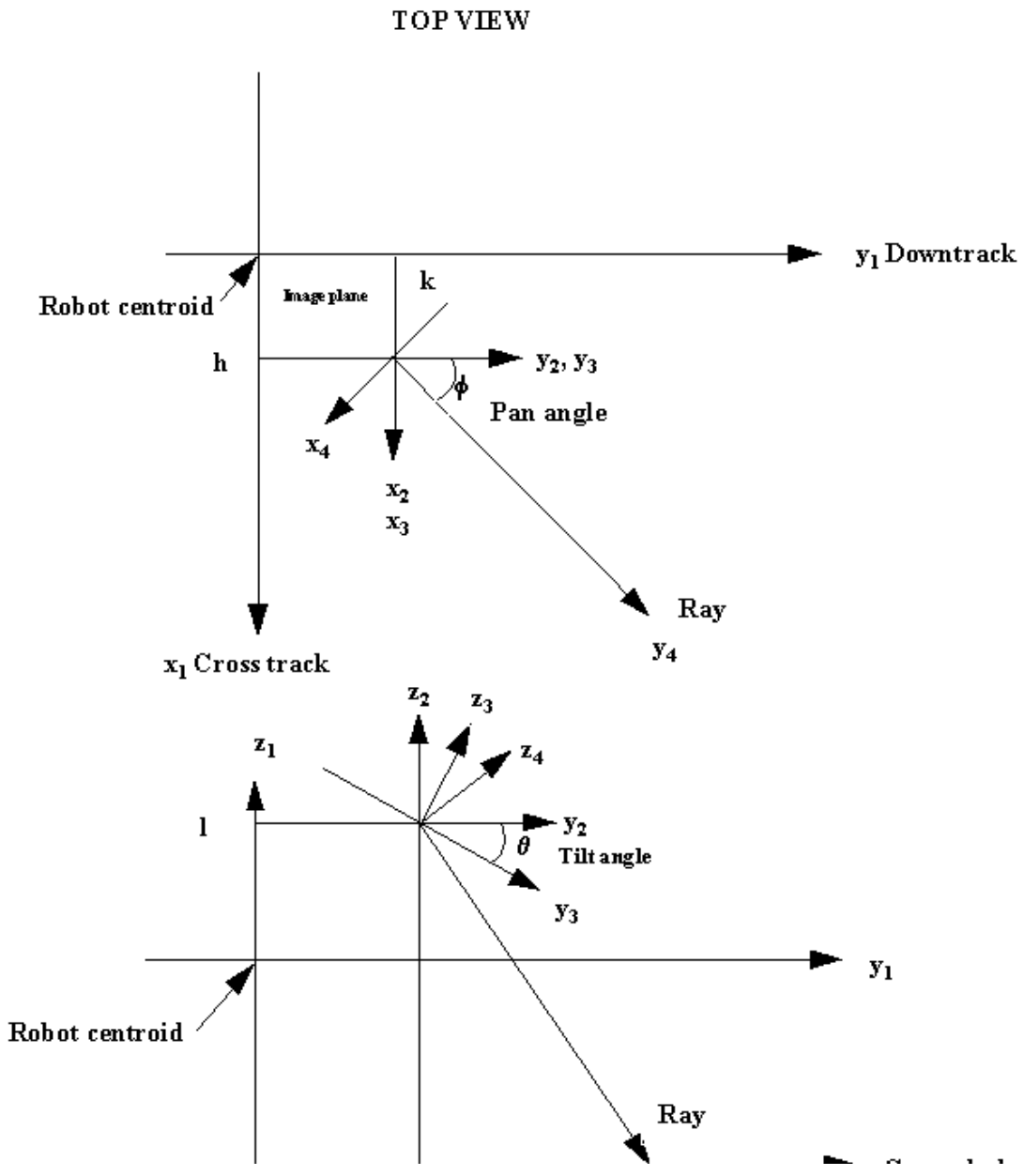
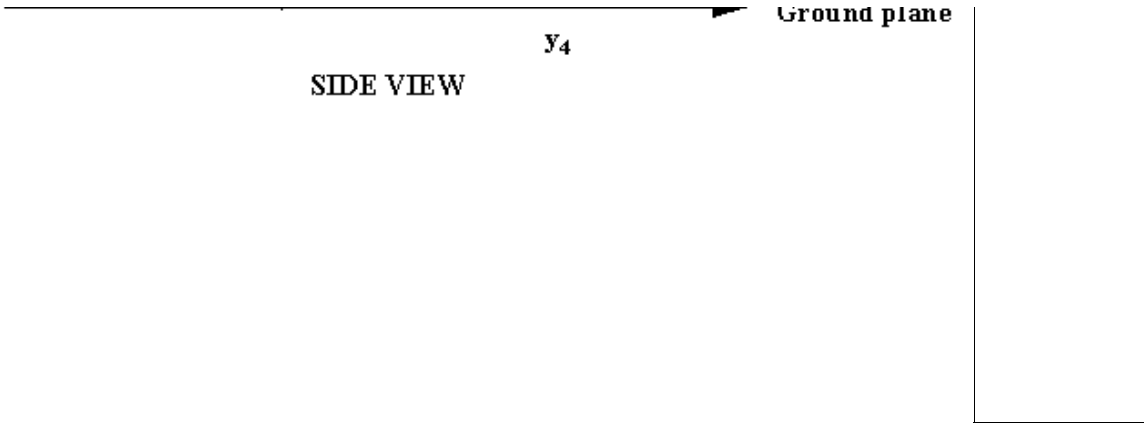



Figure 2. Imaging system Model

#### 2.3.1.1 Forward Transform

The forward transformat is developed using homogeneou coordinates.




### 2.3.1.2 Inverse Transformation

The inverse transformation equations were developed using the ground plane constraint. That is three-dimensional measurements are made by back projecting and intersecting an image ray with the ground plane. The equations were developed symbol ically using the computer program Mathcad. The equations permit computation of ground points from the image points.

### 2.3.2 Determining Model Parameters - System Identification

To make use of the forward and inverse transformations, the system parameters must be determined. The above equations (1-6) have six explicit parameters. Three are the translation locations of the center of the image plane (H, K, and L). Two oth ers are the pan, $\theta$, and tilt angles, $\phi$, of the camera. the sixth parameter is the lens center distance of the camera lens. Direct measurement can be used to give approximat ions to the parameters; however, the parameters (H, K, L) are measured to the center of the image plane while the others are measured at the lens center. Since the image plane is internal to the camera and the lens center is internal to the lens, accurat e direct measurement is difficult. There are also implicit parameters in the actual system. For example the scaling between the optical image and the digital image must be determined. The digital image has a certain number of pixels in the horizontal a nd vertical directions. The camera sensor also has certain physical dimensions.

### 2.3.2 Stereo Vision Principles Model

The stereo vision principle approach [2-propo] uses the scaling between the two coordinate systems to determine the relationship between the physical and image coordinates. The model equation relating the two coordinates is:

$$WxPI = A_{11}x_g + A_{12}y_g + A_{13}z_g + A_{14} \quad \text{.....(7)}$$

$$WyPI = A_{21}x_g + A_{22}y_g + A_{23}z_g + A_{24} \quad \text{.....(8)}$$

$$W = A_{31}x_g + A_{32}y_g + A_{33}z_g + A_{34} \quad \text{.....(9)}$$

where W is the scaling, $A_{nm}$ are coefficients, xPI and yPI are x and y image coordinates, and $x_g$, $y_g$, and $z_g$ are the ground coordinates. Eliminating the scaling, W, and utilizing six matching calibration point s compute the coefficients for the above equations. Below are the resulting two equations:

$$A_{11}x_g + A_{12}y_g + A_{13}z_g + A_{14} - A_{31}x_g xPI - A_{32}y_g xPI - A_{33}z_g xPI = 0 \quad \text{.....(10)}$$

$$A_{21}x_g + A_{22}y_g + A_{23}z_g + A_{24} - A_{31}x_g yPI - A_{32}y_g yPI - A_{33}z_g yPI = 0 \quad \text{.....(11)}$$

The above equations represent two equations in twelve unknown coefficients, $A_{nm}$. The coefficients are computed using magic matrix techniques by utilizing six matching calibration points. Since we are dealing with homogeneous system, the ma trix will include an arbitrary scale factor. If the coefficient $A_{34}$ is set as unity, the resulting transformation matrix will be normalized. With six calibration data points and $A_{34} = 1$, the following matrix equation was formulated .

$$QA = 0 \quad \text{.....(12)}$$

Where

$$Q = \begin{bmatrix} xg1 & yg1 & zg1 & 1 & 0 & 0 & 0 & 0 & -xg1.xPI1 & -yg1.xPI1 & -zg1.xPI1 & -xPI1 \\ 0 & 0 & 0 & 0 & xg1 & yg1 & zg1 & 1 & -xg1.yPI1 & -yg1.yPI1 & -zg1.yPI1 & -yPI1 \\ xg2 & yg2 & zg2 & 1 & 0 & 0 & 0 & 0 & -xg2.xPI2 & -yg2.xPI2 & -zg2.xPI2 & -xPI2 \\ 0 & 0 & 0 & 0 & xg2 & yg2 & zg2 & 1 & -xg2.yPI2 & -yg2.yPI2 & -zg2.yPI2 & -yPI2 \\ xg3 & yg3 & zg3 & 1 & 0 & 0 & 0 & 0 & -xg3.xPI3 & -yg3.xPI3 & -zg3.xPI3 & -xPI3 \\ 0 & 0 & 0 & 0 & xg3 & yg3 & zg3 & 1 & -xg3.yPI3 & -yg3.yPI3 & -zg3.yPI3 & -yPI3 \\ xg4 & yg4 & zg4 & 1 & 0 & 0 & 0 & 0 & -xg4.xPI4 & -yg4.xPI4 & -zg4.xPI4 & -xPI4 \\ 0 & 0 & 0 & 0 & xg4 & yg4 & zg4 & 1 & -xg4.yPI4 & -yg4.yPI4 & -zg4.yPI4 & -yPI4 \\ xg5 & yg5 & zg5 & 1 & 0 & 0 & 0 & 0 & -xg5.xPI5 & -yg5.xPI5 & -zg5.xPI5 & -xPI5 \\ 0 & 0 & 0 & 0 & xg5 & yg5 & zg5 & 1 & -xg5.yPI5 & -yg5.yPI5 & -zg5.yPI5 & -yPI5 \\ xg6 & yg6 & zg6 & 1 & 0 & 0 & 0 & 0 & -xg6.xPI6 & -yg6.xPI6 & -zg6.xPI6 & -xPI6 \\ 0 & 0 & 0 & 0 & xg6 & yg6 & zg6 & 1 & -xg6.yPI6 & -yg6.yPI6 & -zg6.yPI6 & -yPI6 \end{bmatrix}$$

and

$$A = \begin{bmatrix} A_{11} \\ A_{12} \\ A_{13} \\ A_{14} \\ A_{21} \\ A_{22} \\ A_{23} \\ A_{24} \\ A_{31} \\ A_{32} \\ A_{33} \\ A_{34} \end{bmatrix}$$

There are 12 unknowns in the matrix equation. However, since the matrix equation is homogeneous, there is an arbitrary scale. Moving the last column in the matrix Q to the right-hand side and applying the least square regression method can solve the transformation coefficients. Therefore, one coefficient can be arbitrarily selected leaving 11 coefficients to be determined. Since each image point (x, y) gives two equations, a minimum of five and one half image points could give a solution. A gr eater number of points permit a least square solution. After the coefficients are determined, W is computed and for any image coordinate, xPI and yPI, the corresponding ground coordinates are computed.

### 2.3.3 Direct Coefficients Computation Approach

The vision system was modeled by the following equations.

$xPI = A_{11} x_g + A_{12} y_g + A_{13} z_g + A_{14}$ .....(13)

$yPI = A_{21} x_g + A_{22} y_g + A_{23} z_g + A_{24}$ .....(14)

Where $A_{nm}$ are coefficients, xPI and yPI are x and y image coordinates, and $x_g$, $y_g$, and $z_g$ are the ground coordinates. In transforming the ground coordinate points to the image coordinate points the fol lowing transformation operations occur on the points: scaling, translation, rotation, perspective, and projective. Solving for the transformation parameters to obtain the image and ground coordinate relationship is a difficult task as noted in section 3. 2. Fortunately, in the model equations given above, the transformation parameters are imbedded into the coefficients. To compute the coefficients, a calibration device was constructed to obtain 12 data points. With the 12 points, a matrix equation was yielded as shown below:

$XPI = CA_{1k}$ .....(15)

$YPI = CA_{2k}$ ... .....(16)

where

$$C = \begin{bmatrix} xg_1 & yg_1 & zg_1 & 1 \\ xg_2 & yg_2 & zg_2 & 1 \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ xg_{12} & yg_{12} & zg_{12} & 1 \end{bmatrix}; XPI = \begin{bmatrix} xPI_1 \\ xPI_2 \\ . \\ . \\ . \\ xPI_{12} \end{bmatrix}; YPI = \begin{bmatrix} yPI_1 \\ yPI_2 \\ . \\ . \\ . \\ yPI_{12} \end{bmatrix}; A_{1k} = \begin{bmatrix} A_{11} \\ A_{12} \\ A_{13} \\ A_{14} \end{bmatrix}; A_{2k} = \begin{bmatrix} A_{21} \\ A_{22} \\ A_{23} \\ A_{24} \end{bmatrix}$$

Equations (3) and (4) consist of 12 linearly independent equations and four unknowns, the least-square regression method is applied to yield a minimum mean-square error solution for the coefficients. Below are the equations for the solution:

$A_{1K} = (C^TC)^{-1}C^TXPI$ .....(17)

$A_{2K} = (C^TC)^{-1}C^TYPI$ .....(18)

Given an image coordinate xPI and yPI, and z ground coordinate (the z coordinate of the points with respect to the centroid of the robot is maintained constant because of the ground constraint) the corresponding $x_g$ and $y_g$ grou nd coordinates are computed as indicated by the following matrix equations.

$$\begin{pmatrix} x_g \\ y_g \end{pmatrix} = Q^{-1}B$$ .....(19)

where

$$Q = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}; B = \begin{bmatrix} xPI - A_{14} - (A_{13}z_g) \\ yPI - A_{24} - (A_{23}z_g) \end{bmatrix}$$

Note that equation (1) and (7) can be modified to accommodate the computation of $z_g$ when an elevation of the ground surface is considered.

## 3.0 Experimental Methods

A calibration device was constructed to permit measurement of corresponding three dimensional object points and image points. The determination of the camera focal lengths and the orientation of the projection system (system identification) w ith respect to the global coordinates system can be obtained by several methods as described in the introduction. In this study, a calibration device was constructed to accomplish system identification.

### 3.1 Calibration Device

Figure 3 shows the calibration device lying on a scaled graph sheet. The device comprises a wooden base, painted black for contrast, six white ping-pong balls, and two five inch long poles. The wooden base is 14.5" x 11.5" x 0.75&qu ot;. Four of the balls are placed on the same plane (the surface) of the wooden base while the poles that are pinned to the wooden base support the other two balls. Six other darkened points are spotted on the graph sheet, bringing the total points cons idered for the calibration to twelve. The points on the graph sheet are considered to be on the ground level.

Starting from the surface of the wooden base, the balls are given alphabetic labels in a counter clockwise direction. Each of the corners of the wooden base is numerically labeled, also in a counter clockwise direction, with the corner #1 coinciding w ith ball A.

Accurate measurements of the exact coordinates of all the twelve points with respect to the reference point are an essential factor in the calibration process. To attain the needed high accuracy, a coordinate measuring machine, whose accuracy is a bout ten thousandth of an inch, was utilized to measure the centroid of the six balls with respect to the tip of corner #1 of the wooden base. The darkened points on the graph sheet are also precisely located with respect to the tip of corner #1. To obt ain the actual physical measurements of each of the twelve calibration points with respect to a reference point, in this case the centroid of the robot, the X and Y distances between the tip of corner #1 of the wooden base and the centroid of the robot is carefully and painstakingly measured.

Each point of the calibration system now has an accurate physical coordinate reference to

the centroid of the robot and their corresponding image coordinates are obtained via Iscan, the image-processing tool. From the physical and image coordinate s, the camera parameters (coefficients) are computed.

### 3.2 Results of the Calibration Process

In section 3, the mathematical aspect of the calibration system was discussed.

Three different models were considered. The results for each mathematical model will be discussed in sequel.

#### 3.2.1 The homogeneous Matrix Transformation

In the sub-section 3.3.2 it was noted that the six parameters associated with this model are difficult to measure. This is due to the fact that the measurement is made in reference to the center image plane and lens center which is internal to the lens and inaccessible. For these reasons, arbitrary values were assumed for the parameters. These values were iteratively changed in order to minimize the mean square error between the model calculated data points and the measured calibration data. Figure 4 shows the data point plots for the model and calibrations.

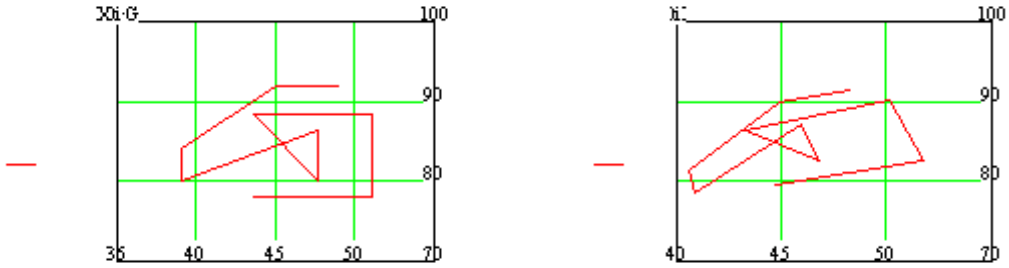



Figure 4. Data points for evaluating the homogeneous transformation model

Despite the fact that the parameters chosen gave the least minimum mean square error, the deviation (error) of the model points from that of the calibration is too high to be neglected. This discrepancy was due to lack of direct access to the camera l ens which made almost impossible to measure the required parameters; as such the forward and inverse homogeneous transformation model is being given further investigation and was not implemented to solve the vision problem.

#### 3.2.2 Stereo vision principles model

This model involved the computation of the coefficients and the scale factor, W. Because the number of unknowns in equations 10 and 11 is more than the number of equations, there are infinitely many solution to this model. Even with the gener ated matrix equations from the calibration data, one of the values for the coefficients had to be assumed, in this case it was assumed to be 1. A set of calibration data points is shown in Table 1. The computed coefficients are indicated in Table 2. Fig ure 5 shows the data point plots for the model and calibrations.

Table 1. Set of Calibration Data Points

| Points | Physical Coordinates | | | Image Coordinates | |
|---|---|---|---|---|---|
| | x | y | z | x | y |
| 1 | 56.481 | 74.547 | -22.533 | 197 | 160 |
| 2 | 63.971 | 74.554 | -22.524 | 316 | 187 |
| 3 | 63.971 | 85.048 | -22.564 | 263 | 86 |
| 4 | 56.456 | 85.083 | -22.552 | 152 | 54 |
| 5 | 60.544 | 76.609 | -19.057 | 303 | 125 |
| 6 | 60.495 | 83.0375 | -19.034 | 270 | 62 |

Table 2. Camera Parameters

| Coefficients | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| A11 | A12 | A13 | A14 | A21 | A22 | A23 | A24 | A31 | A32 | A33 | A34 |
| -8.603 | 0.125 | -2.21 | 358.454 | -2.78 | 3.198 | 4.363 | -39.689 | -0.009 | -0.008 | 0.011 | 1 |

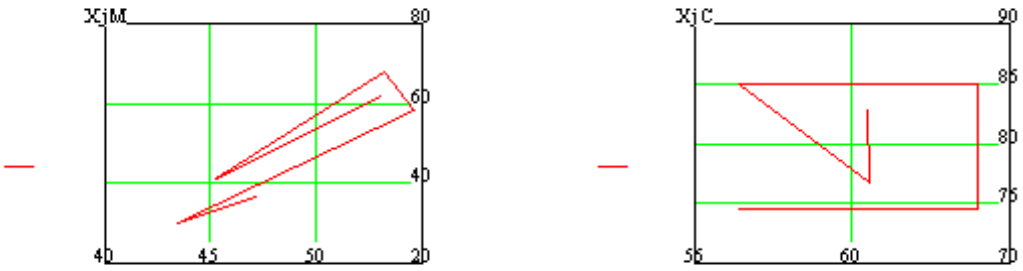



Figure 5. Data points for evaluating the stereo vision principle model

The results of this model are way off the mark. As demonstrated by the plot in Figure 5, the model points have been scattered in disarray. The accuracy of this model was dependent on the coefficients. However because there are infinitely many soluti on to the matrix equation, arriving at the best solution is extremely difficult if not impossible. This model is being given further thoughts and was not implemented to solve the vision problem.

### **3.2.3** Direct **coefficient computation approach**

Unlike the stereo vision principles model where the scaling was expressed as a discrete equation, this model treated all the transformational operations, including scaling to be embedded into the coefficients. To solve for the coefficients, th e following set of calibration data points was utilized. Table 3 has the data points while the camera parameters are indicated in Table 4. Figure 6 shows the data point plots for the model and calibrations.

Table 3. Set of Calibration Data Points

| Points | Physical Coordinates | | | Image Coordinates | |
|---|---|---|---|---|---|
| | x | y | z | x | y |
| 1 | 56.481 | 74.547 | -22.533 | 197 | 160 |
| 2 | 63.971 | 74.554 | -22.524 | 316 | 187 |
| 3 | 63.971 | 85.048 | -22.564 | 263 | 86 |
| 4 | 56.456 | 85.083 | -22.552 | 152 | 54 |
| 5 | 60.544 | 76.609 | -19.057 | 303 | 125 |
| 6 | 60.495 | 83.0375 | -19.034 | 270 | 62 |
| 7 | 53.375 | 72.5 | -24 | 134 | 182 |
| 8 | 53.375 | 82.5 | -24 | 94 | 74 |
| 9 | 57.375 | 86.5 | -24 | 143 | 55 |
| 10 | 62.375 | 86.5 | -24 | 218 | 77 |
| 11 | 67.375 | 82.5 | -24 | 305 | 131 |
| 12 | 67.375 | 72.5 | -24 | 356 | 230 |

Table 4. Camera Parameters

| | | | Coefficients | | | | |
|---|---|---|---|---|---|---|---|
| A11 | A12 | A13 | A14 | A21 | A22 | A23 | A24 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 15.4 | -4.353 | 14.213 | -27.405 | 3.807 | -9.919 | -7.895 | 505.123 |

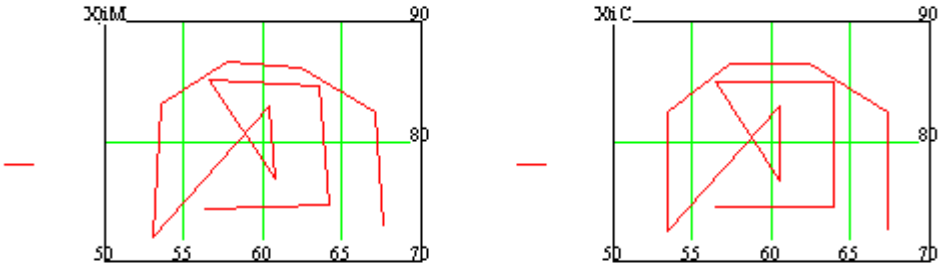



Figure 6. Data points for evaluating the direct coefficient computation model

The deviation of the model points from the calibration points shown in Figure 6 is very minimal compared to the plots in Figures 4 and 5. With an accuracy of about tenth of an inch, this model has proven to be the best among the three models considere d in this study. As a result of this reliable performance, the direct coefficient computation model was thus implemented to solve the vision problem.

## 4.0 Experimental Results for the Vision System

The computation of the camera parameters sets the stage for computing the physical coordinates given any image coordinate. To show how the physical coordinates are computed given any image coordinate, another calibration was performed to obtai n camera parameters. As discussed in the previous chapter, the mathematical model implemented henceforth is the direct coefficient computation method. The previous equations were used in the computation of the physical coordinates. Note that the z-coord inate for each of the points was treated as constant because in the real time implementation of this method, the z-coordinate is constrained by the ground. Table 5 shows the calibration parameters and the results of the physical coordinate computations ar e shown in Figure 6.

**Table 5. Camera Parameters**

| | | | Coefficients | | | | |
|---|---|---|---|---|---|---|---|
| A11 | A12 | A13 | A14 | A21 | A22 | A23 | A24 |
| 14.7 | -11.501 | -1.189 | 415.456 | -3.567 | -5.521 | -9.944 | 559.417 |

**Table 6. Set of Original and Computed Calibration Data Points**

| Points | Original Physical Coordinates | | | Image Coordinates | | Computed Physical Coordinates | |
|---|---|---|---|---|---|---|---|
| | x | y | z | x | y | x | y |
| 1 | 48.856 | 71.047 | -22.533 | 341 | 220 | 48.464 | 70.745 |
| 2 | 56.346 | 71.054 | -22.524 | 454 | 187 | 56.667 | 71.404 |
| 3 | 56.346 | 81.548 | -22.564 | 329 | 135 | 55.949 | 81.358 |
| 4 | 48.831 | 81.583 | -22.552 | 219 | 158 | 48.803 | 81.789 |
| 5 | 52.919 | 73.109 | -19.057 | 377 | 153 | 53.33 | 73.475 |
| 6 | 52.87 | 79.537 | -19.034 | 301 | 124 | 52.605 | 79.154 |
| 7 | 44.55 | 72 | -24 | 265 | 245 | 43.97 | 71.762 |
| 8 | 47.75 | 72 | -24 | 330 | 229 | 48.414 | 71.789 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 9 | 46.75 | 74.5 | -24 | 265 | 221 | 46.229 | 74.649 |
| 10 | 46 | 79 | -24 | 215 | 195 | 46.417 | 79.236 |
| 11 | 49.75 | 82 | -24 | 236 | 168 | 49.908 | 81.871 |
| 12 | 40.15 | 83 | -24 | 81 | 195 | 40.362 | 83.149 |

Correlation plots for the original and the computed x and y coordinates are shown in figures 7 and 8. The linearity of the plots means that the difference between the original coordinates and the computed ones is very small. Also computed to asce rtain or test the discrepancies between the two sets of coordinates is the mean square error. For each of the correlation plots, the mean square error was 0.242 for the x-axis and 0.295 for the y-axis. With a mean square error of within tenth of an inch, the calibration process is considered as accurate and reliable enough to compute the physical coordinate of a real life point on a ground. Having dealt with the computation of the x and y coordinate of a physical point with respect to the centroid of th e robot, the next item considered in this study is how to spot more than one point, two points to be precise, on a line and establish some geometrical relationship between the points and relate it to the centroid of the robot. In the next section, the de termination of this relationship and how it is utilized to control the steering of the robot to follow the line is discussed.

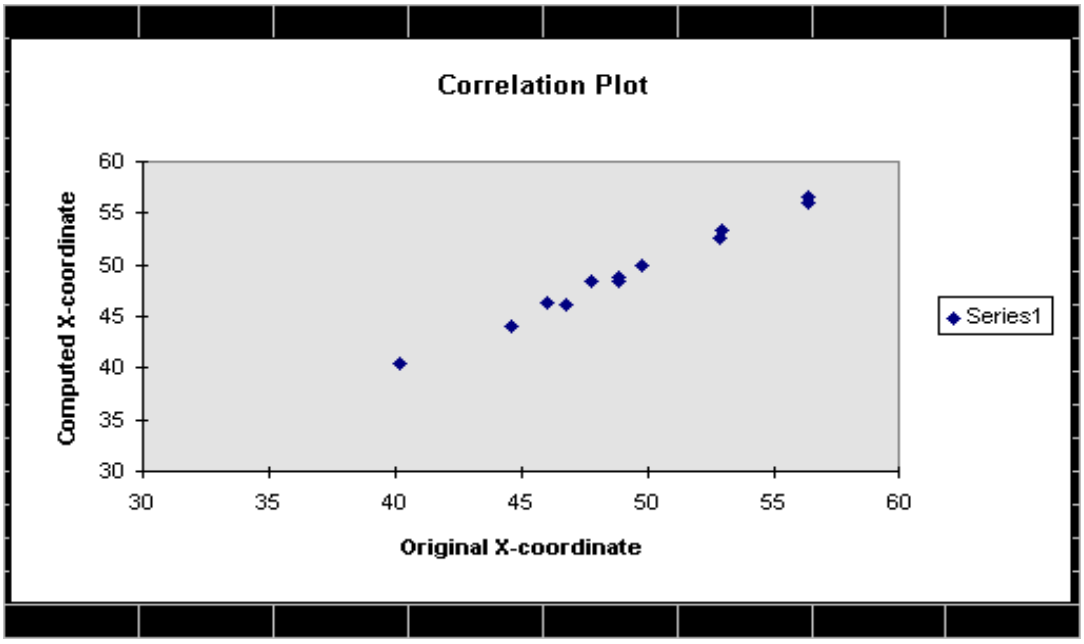



Figure 7. Correlation plot for the X-axis. (Mean square error = 0.242)

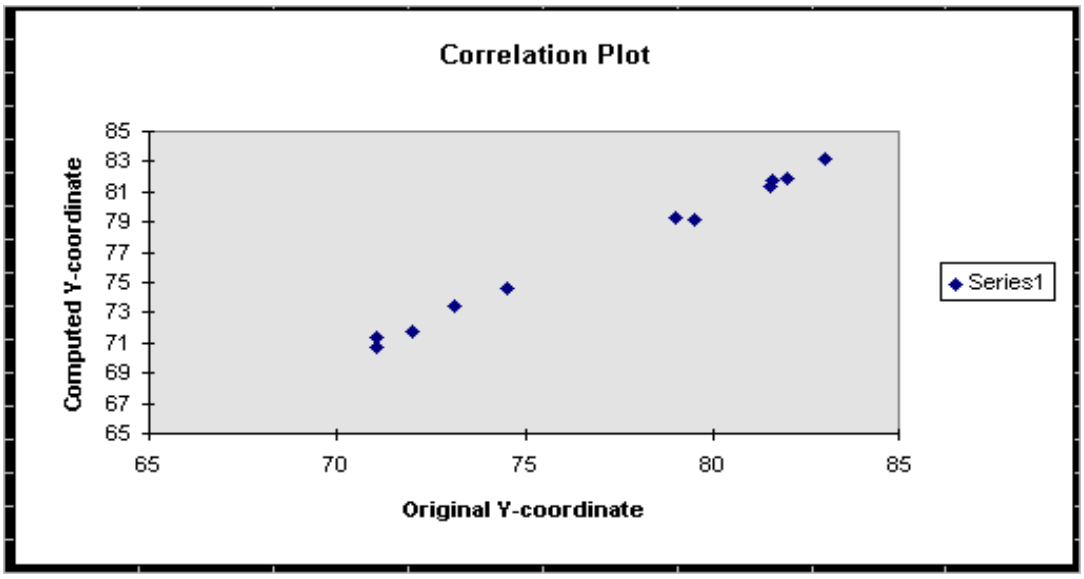



Figure 8. Correlation plot for the Y-axis. (Mean square error = 0.295)

## 5.0 Vision Guidance - Control System

The image processing of the physical points is done by the Iscan tracking device [14], which returns the centroid of the brightest or darkest region in a computer controlled windows and returns its X and Y coordinates. Two points on the line a re windowed and their corresponding coordinates are computed as described above. From the computed x and y ground coordinates of the points, the angle of the line with respect to the centroid of the robot is computed from simple trigonometric relationshi p. In the next section, we shall show how the angle of the line just computed is used with other parameters to model the steering control of the robot using a look-up table.

### 5.1 Inputs to the controller

The mobile robot is designed to maneuver an obstacle course. The width of the course is 10 feet. In controlling the robot to follow the obstacle course, the angle of the line with respect to the centroid of the robot and the angle of the fro nt wheel with respect to the centroid of the robot along the direction of motion (y-axis) are considered. Also considered is the perpendicular distance of the line to the centroid with respect to the x-axis. Both the distance and the angle errors need t o be minimized. The figure [15] shows why both parameters should be controlled. If only the angle is controlled, the robot might draw closer to the line and eventually go out of bounds and the same is true for minimizing only the distance error.

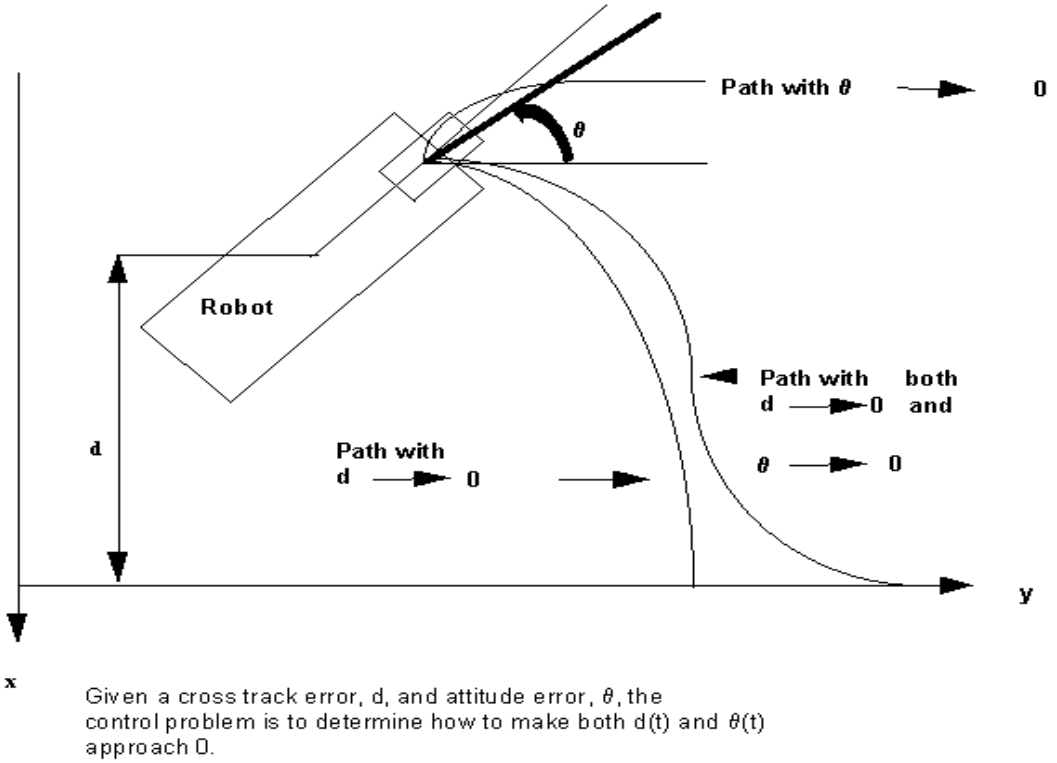



Figure ??. Depiction of why both d and θ must be controlled

From vision system, two points on the line are windowed. The physical coordinates of these points with respect to the robot's centroid are computed using the mathematical model and the calibration process discussed under the vision system. Using a geometrical relationship, the angle of the line with respect to the centroid of the robot is computed. Let $X_1$ and $Y_1$ be the coordinate for the first point and let $X_2$ and $Y_2$ be the coordinate for the second poi nt. The angle of the line $\theta_{line}$ is computed as

$\theta_{line} = \tan^{-1} (X_2\text{-}X_1/Y_2\text{-}Y_1)$ ???

Computing the distance of the line to the centroid of the robot with respect to the x-axis was not an obvious task because there are many directions that the line can orient to with respect to the robot. Using the perpendicular distance computation me thod for computing the shortest distance of a line to a point would give a false distance between the line and the robot's centroid because of the different orientations. Several other methods were considered but were not implemented because of the same reason as that of the perpendicular method. The method that was employed was finding the mid-point of the line

(formed by the two points under consideration) and letting the x-coordinate of the mid-point represent the distance of the line to the centroid of the robot. This distance is denoted by $d_r$.

To develop a control system to guide the robot, the essential inputs as discussed above are the angle error $\theta_{error}$ and the distance $d_{error}$. The $\theta_{error}$ is the difference between the angle of the line and that of the front wheel with respect to the centroid of the robot. The angle of the wheel is denoted by $\theta_w$ and it is obtained via the encoder. The output of the control system is the steering angle, $\theta_{steering}$. The convention for steering the wheel is that positive angles are to the right and negative angles to the left.

Four possible cases are considered in computing the $\theta_{error.}$ The cases are:

1. If angle of the wheel,$\theta_w$, is positive while the angle of the line,$\theta_{line}$ is negative then the error angle is computed as:

$$\theta_{error} = -(\theta_w + |\theta_{line}|) \quad (??)$$

The steering wheel needs to be turned to the left; i.e. $\theta_{error}$ has a negative value.

2. If $\theta_w$ is positive, and $\theta_{line}$ is also positive, the error angle is:

$$\theta_{error} = \theta_{line} - \theta_w \quad (??)$$

The direction of the steering is dependent on the sign of $\theta_{error}$.

3. If $\theta_w$ is negative, and $\theta_{line}$ is positive, then the error angle is:

$$\theta_{error} = |\theta_w| + \theta_{line} \quad (??)$$

The sign of $\theta_{error}$ is positive, as such, the direction is right.

4. If $\theta_w$ is negative, and $\theta_{line}$ is also negative, then the error angle is:

$$\theta_{error} = \theta_w + \theta_{line} \quad (??)$$

The direction of steering will depend on the sign of $\theta_{error}$.

The algorithm to calculate $\theta_{error}$ is coded in C++. The steering wheel has a limit of ± 20 degrees turning angle. As a result, $\theta_{error}$ is bounded by ± 20. If the magnitude of $\theta_{error}$ is greater than 20, it is reduced to 20. $\theta_l$ is computed from the vision guidance model discussed in the previous section while $\theta_w$ is obtained from the encoder.

The difference between the required distance from the centroid to the line, $d_r$, and the actual computed distance of the line to the centroid, $d_l$ is $d_{error}$. Given below is the equation:

$$d_{error} = d_r - d_l \quad (???)$$

where $d_r$ is five feet because the width of the obstacle course is ten feet. If $d_{error}$ is negative, it means that the robot is closer to the left line of the course and if $d_{error}$ is positive, the robot is nearer to the right line of the course. Both $d_{error}$ and $\theta_{error}$ are combined by the control system to output a suitable steering angle, $\theta_{steering}$, to guide the robot.

**5.2 Control Method:**

The field of control systems has been in existence for several decades and enormous amount of theories and proven techniques have been devised and implemented on various technological devices such as automobiles, aircraft, health-related instru ments, space vehicles etc. Recently artificial intelligence methods have also been employed to develop control systems. Tanaka presented a fuzzy logic controller that guarantees stability of a control system for a computer-simulated model car in [16]. In [17], Altrock et al discussed advanced fuzzy logic application for automobile application. Samu, Nikhil, and Hall presented a fuzzy logic approach to modeling a control system for a mobile robot using MATLAB simulation software [18].

In this study, a look-up table was created to model the control system. Table ?? shows the look-up table. The first column has the angle error while the first row contains the distance

error. The remaining entries of the table represent the stee ring angles. The steering angle is intuitively and arbitrarily chosen after combining the row and the column of that particular cell of the table. For instance, the steering angle ???, located at row ?? and column ?? is the result of combining the angl e error of column ?? and distance error of row ??.

Table 7. Look-up Table for the control system.

| | | | | Distance Error (inches) | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Angle-Error | -36 | -30 | -24 | -8 | 0 | 8 | 24 | 30 | 36 |
| -20 | **-20** | **-20** | **-20** | **-20** | **-20** | **-15** | **-8** | **5** | **5** |
| -15 | **-18** | **-18** | **-17** | **-16** | **-15** | **-10** | **-5** | **4** | **5** |
| -10 | **-12** | **-12** | **-12** | **-11** | **-10** | **-7** | **-3** | **3** | **5** |
| -5 | **-8** | **-7** | **-6** | **-6** | **-5** | **-4** | **-2** | **2** | **5** |
| 0 | **-5** | **-3** | **-2** | **-1** | **0** | **1** | **2** | **3** | **5** |
| 5 | **-5** | **-2** | **2** | **4** | **5** | **6** | **6** | **7** | **8** |
| 10 | **-5** | **-3** | **3** | **7** | **10** | **11** | **12** | **12** | **12** |
| 15 | **-5** | **-4** | **5** | **10** | **15** | **16** | **17** | **18** | **18** |
| 20 | **-5** | **-5** | **8** | **15** | **20** | **20** | **20** | **20** | **20** |

Table 7 was coded in C++ and implemented on the robot. Appendix A has the source code for this program. This program is the main software program that controls the robot. While executing, the program calls the vision system, the ultrasonic system , the Galil control board, and the emergency systems.

## 6.0 References

[1] P. F. Muir and C. P. Neuman, 'Kinematic Modeling of Wheeled Mobile Robots,' **Journal of Robotic Systems**, 4(2), 1987, pp. 281-340

[2] E. L. Hall, **Robotics: A User-Friendly Introduction**, Holt, Rinehart, and Winston, New York, NY, 1985, pp. 23.

[3] Z. L. Cao, S. J. Oh, and E. L. Hall, "Dynamic omnidirectional vision for mobile robots," **Journal of Robotic Systems**, 3(1), pp. 5-17, 1986.

[4] Z. L. Cao, Y. Y. Huang, and E. L. Hall, "Region Filling Operations with Random Obstacle Avoidance for Mobile Robots," **Journal of Robotics Systems**, 5(2), 1988, pp. 87-102.

[5] S. J. Oh and E. L. Hall, "Calibration of an omnidirectional vision navigation system using an industrial robot," **Optical Engineering**, Sept. 1989, Vol. 28, No. 9, pp. 955-962.

[6] R. M. H. Cheng and R. Rajagopalan, "Kinematics of Automated Guided Vehicles with an Inclined Steering Column and an Offset Distance: Criteria for Existence of Inverse Kinematic Solution," **Journal of Robotics Systems**, 9(8), 1059-108 1, Dec. , 1992.

[7] M. P. Ghayalod, E. L. Hall, F. W. Reckelhoff, B. O. Mathews and M. A. Ruthemeyer, "Line Following Using Omnidirectional Vision," Proc. of SPIE Intelligent Robots and Computer Vision Conf., SPIE Vol. 2056, Boston, MA, 1993.

[8] Kazuo Tanaka, "Design of Model-based Fuzzy Controller Using Lyapunov's Stabilty Approach and its Application to Trajectory Stabilization of a Model Car," **Theoretical Aspects of Fuzzy Control**, John Wiley & sons, Inc, pp.31-50, 1 995. 2nd IEEE conference on fuzzy system, San Francisco, CA, 1993

[9] C. V. Altrock et al., Advanced fuzzy logic control technologies in automotive applications. Proceedings of 1st IEEE international Conference on Fuzzy Systems, pp. 835-842, 1992.

[10] Textron Inc., Owner's Operation and Service Manual, Augusta, Georgia(1987).

[11] Bradley Mathews, Mike Ruthemeyer, David Perdue and Ernest Hall, "Line Following for a Mobile Robot," Proc. of SPIE Intelligent Robots and Computer Vision Conf., SPIE Vol. 2588, Boston, MA, 1995

[12] General Electric, EV-1 SCR Control Manual, Charlottesville, Virginia(1986).

[13] Galil Inc, DMC-1000 Technical Reference Guide Ver 1.1, Sunnyvale, California(1993).

[14] BEI Motion Systems Company, Model H20 Technical Specifications, Goleta, California (1992).

[15] Fauver Corp., Fauver Linear Actuators Catalog, Cincinnati, Ohio(1993).

[16] Reliance Electric, Electro-Craft BDC-12 Instruction Manual, Eden Prairie, Minnesota (1993).

[17] Delco Corp., Motor Specifications, Dayton, Ohio(1993).

Applications:

1. AGV systems in a wharehouse operations- white lines for the lanes replaces underground wiring systems for AGVs,

2. Pickup operations of items by a robot from a conveyor as the items have a fixed height on the conveyor....

3. More.....

References

[1] Hall, E. L. "Fundamental Priciples of Robot Vision" Handbook of Pattern Recognition and Image Processing: Computer Vision. pp. 543-575. 1994

[2] Borenstein, J., et al ""Where am I?" Sensors and Methods for Mobile Robot Positioning," University of Michigan, Chapter 9, pp. 211.

[3] Lovenitti, P., Thompson, W. and Singh, M. "Three-dimesional Measurement using a Sinle Image," SPIE, Opt. Eng. **35**(5) 1496-1502. May 1996.

[4] Hong, F. and Baozong, Y. "An Accurate and Practical Camera Calibration System for 3D Computer Vision" Chinese Journal of Electronics. Vol. 1, No. 1, pp. 63-71, June 1991

[5] F. I. Parke, "Measuring Three-Dimensional Surfaces with aTwo-Dimensional Data Tablet," Computer and Graphics, Vol. 1, 1975, pp. 5-7

[6] W. D. Renner, et al., "The Use of Photogrammetry in Tissue Compensation Design," Radiology, Vol. 125, November, 1977, pp. 505-510.

[7] Tio, J. B. "Single Image Surface Measurement for Robotics Application" Masters thesis, University of Tennessee, Knoxville, August, 1982, pp. 20.

[8] Tsai, R. Y., "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-The-Shelf Cameras and Lenses." IEEE Transaction on Robotics and Automation, vol. 8, no. 2, pp. 129-139.

[9] Liu, Y., Huang, T. S., and Faugeras, O. D., "Determination of Camera Location from 2-D to 3-D Line and Point Correspondence." IEEE transaction on Pattern Analysis and Machine Intelligence, Vol. 12, no. 1, pp. 28-37, 1990.

[10] Haralick R. M. et al., "Pose Estimation from Corresponding Point Data." IEEE Transactions on Systems, Man, and Cybernetics, vol 19, no. 6, pp. 1426-1445. 1989

[11] Kumar, "Determination of the Camera Location and Orientation." Proceedingss of Image Understanding Workshop 88, pp. 870-881, 1988.

[12] Yuan, J. S. C., "A General Photogrammetric Method for Determining Object Position and Orientation." IEEE Transaction on Robotics and Automation, vol. 5, no. 2, pp. 129-142. 1989.

[13] Chen, H. H., "Pose Estimation from Line-to-Plane Correspondences." IEEE Transaction on Pattern Analysis and Machine Intelligence, vol 13, no. 6, pp. 530-541.

[14] Iscan Inc., RK-446-R Video Tracking System Manual, Cambridge, Massachusets, 1993.

[16] Kazuo Tanaka, "Design of Model-based Fuzzy Controller Using Lyapunov's Stabilty Approach and Its Application to Trajectory Stabilization of a Model Car," **Theoretical**

**Aspects of Fuzzy Control**, John Wiley & sons, Inc, pp.31-50, 1995. 2nd IEEE conference on fuzzy system, San Francisco, CA, 1993

[17] C. V. Altrock et al., Advanced fuzzy logic control technologies in automotive applications. Proceedings of 1st IEEE international Conference on Fuzzy Systems, pp. 835-842, 1992.

[18] Samu, T. I.., Kelkar, N., and Hall, E., "Fuzzy Logic System for Three Dimensional Line Following for a Mobile Robot," Proc. Adaptive Distributive Parallel Computing Symposium, Dayton, Oh., pp. 137-148.