

Visual Servoing of an Omni-Directional Mobile Robot for Alignment with Parking Lot Lines

Matthew Berkemeier Morgan Davidson Vikas Bahl Yangquan Chen Lili Ma
Center for Self-Organizing and Intelligent Systems and
Dept. of Electrical & Computer Engineering
Utah State University, Logan, UT 84322

Abstract—ODIS is an omni-directional mobile robot designed to autonomously or semi-autonomously inspect automobiles in a parking lot. Periodically, its position and orientation references need to be reset. This paper considers visual servoing to parking lot lines as one possible approach. Analysis and simulations demonstrate that a surprisingly simple proportional controller in the image coordinates can accomplish position and orientation alignment with parking lot lines. Unlike previous work, no image Jacobian matrix is necessary. Knowledge of the camera focal length is not required, but the camera and vehicle axes are assumed to be aligned, and the vehicle is assumed to rotate about the camera frame's y -axis.

I. INTRODUCTION

THE Utah State University Omni-Directional Inspection System (ODIS) is a small mobile robotic system that can be used for autonomous or semi-autonomous inspection under vehicles in a parking lot [1], [2]. Customers for such a system include military police and other law enforcement and security entities [3]. The robot features (a) three “smart wheels” [4], [5] in which both the speed and direction of the wheel can be independently controlled through dedicated processors, (b) distributed computing over 8 processors, (c) a sensor array with a laser, sonar and IR sensors, and a video camera. The “smart wheel” [4] is a unique feature in ODIS which was developed by the Center for Self-Organizing and Intelligent Systems (CSOIS). This has resulted in the T-series of omni-directional (ODV) robots [5]. With the ODV approach, the CSOIS robots, including ODIS, can achieve complete control of orientation and motion in a plane, thus making the robots holonomic. Fig. 1 shows the ODIS design.

ODIS was designed to operate for an extended period outdoors and will periodically need to reset its fiber-optic gyro and position references. Since the robot will be operating in a parking lot, lines, which serve as guides for automobile parking, are natural references for both position and orientation. ODIS is equipped with a color camera for inspection tasks, so visual servoing to parking lot lines seems like a natural

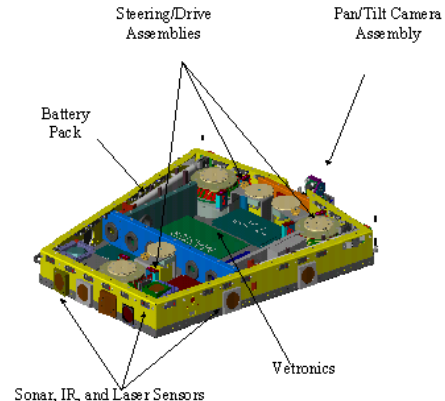


Fig. 1. ODIS Design.

fit. The purpose of this paper is to study a very simple algorithm for aligning ODIS to a line.

Hutchinson, Hager, and Corke [6] reviewed previous work in visual servoing and provided a tutorial introduction to standard methods. Visual servoing approaches were divided into 4 categories. In this paper a “dynamic image-based look-and-move” approach is used. It is “image based” since errors are computed in the image. It is “dynamic look-and-move” since the errors determine corrective velocities for the robot, which must then be then fed to a lower-level robot velocity controller. In contrast to most image-based approaches, the method here does not require the use of an image Jacobian matrix. The image Jacobian matrix maps robot velocities to image feature velocities. Since the inverse map is needed in a robot controller, the image Jacobian must usually be inverted somehow. Instead of using this technique, the approach in this paper involves a simple proportional controller to map image errors to vehicle velocities. Although the image errors are nonlinearly related to the vehicle position errors, with significant coupling between degrees of freedom, convergence to the desired position and orientation occurs for virtually any initial condition of interest.

Much of the previous work has focused on servoing

to point features. In contrast, the approach discussed in this paper makes use of lines. Some examples of previous papers, which have considered servoing to lines, are discussed next.

Espiau, Chaumette, and Rives [7] considered a very general version of the visual servoing problem. Servoing to lines, as well as points, planes, circles, spheres, etc. were all discussed. In contrast to this paper, the image Jacobian matrix was an important part of their control method (However, they used the term “interaction matrix” instead of image Jacobian matrix).

Hager [8] used Plücker coordinates for lines. Both points and lines were used as features in the visual servoing problem. Stereo vision was assumed. He also used an inverse image Jacobian matrix to accomplish the mapping from image errors to manipulator velocities. In this paper, lines are assumed to lie in a plane (the parking lot) parallel to the camera’s x - z plane. Specifically, a line can be represented by a point (x, z) in the parking lot plane ($y = -c < 0$) and an angle (ψ) . This is much less general but allows a simpler representation than that used in [8].

Andreff, Espiau, and Horaud [9] discussed the problem of aligning a desired image of lines with an image containing lines. They used a similar but slightly different representation of lines than [8] (normalized Plücker coordinates). Instead of an image-based approach, they recovered the line parameters in the camera frame coordinates, and used this to drive the error in alignment to zero.

Madsen and Christensen [10] presented an algorithm to direct a manipulator-mounted camera to line up with the normal to a plane formed by two polyhedral edges. It was based on the angle between the image of the two edges. The mobile robots in the work of Castellanos and Tardós [11] made use of lines in the camera image to locate themselves and to build maps, but no visual servoing was performed.

A representative example of visual servoing for mobile robots is the work of Burschka and Hager [12]. They considered path tracking of a nonholonomic mobile robot. Points (not lines) were the feature servoed in the image. Like most of the image-based work (but in contrast to the approach in this paper), an inverse image Jacobian matrix was used to relate image errors to commanded vehicle velocities.

The paper is organized as follows: Section II presents the equations for the models of the camera, vehicle kinematics, and proportional controller. Section III offers 2 results on the stability of the proportional controller. Section IV presents a representative simulation result. Section V offers some possible extensions of the approach which will be considered in the near future. Finally, conclusions are briefly dis-

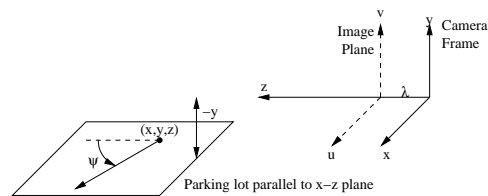


Fig. 2. Camera Frame, Image Plane, and Parking Lot Line.

cussed in Section VI.

No experimental results are reported in this paper. However, some preliminary results have been reported in [13], which focused on ODIS hardware and software.

II. THE MODEL

A. Perspective Map

A perspective projection model is assumed for the camera. The projection map is given by

$$(u, v) = \frac{\lambda}{z}(x, y). \quad (1)$$

Note that, under perspective projection, straight lines in the camera frame map to straight lines in the image plane.

Fig. 2 illustrates the conventions for the camera frame, image plane, and parking lot line. Consider a point in the camera frame with coordinates (x, y, z) . Now, add the unit vector $(\sin \psi, 0, \cos \psi)$ to this point to obtain a second point. It will be assumed that $-\pi/2 < \psi < \pi/2$. The angle that the image of this vector makes with respect to the positive v -axis in the image can be easily derived.

The mappings of the 2 points, under the perspective projection are given by

$$\begin{aligned} (x, y, z) &\mapsto \frac{\lambda}{z}(x, y) \\ (x + \sin \psi, y, z + \cos \psi) &\mapsto \frac{\lambda}{z + \cos \psi}(x + \sin \psi, y). \end{aligned}$$

Taking the difference of the 2 image points gives a vector in the image plane:

$$\frac{\lambda}{z(z + \cos \psi)}(z \sin \psi - x \cos \psi, -y \cos \psi).$$

Note that the vector always has a positive v component since $y < 0$ (the camera is above the parking lot). The angle of this vector with respect to the positive v -axis is then given by

$$\theta = \sin^{-1} \left(\frac{z \tan \psi - x}{\sqrt{(z \tan \psi - x)^2 + y^2}} \right), \quad (2)$$

where the angle is positive if the vector has a positive u component. Note that, based on Fig. 2, the image

plane should be visualized with u positive in the left direction, and positive θ is then in the counter-clockwise direction.

B. Proportional Controller

Assume an image processing algorithm is available to extract the starting point and angle of the image of a parking lot line. Such an algorithm has been previously developed by researchers in the CSOIS.

A simple controller (based on intuition) is given by

$$\dot{x}_v = k_1(u - u_d) \quad (3)$$

$$\dot{z}_v = k_2(v - v_d) \quad (4)$$

$$\dot{\psi}_v = k_3(\theta - \theta_d), \quad (5)$$

where k_i , $i = 1, 2, 3$ are positive controller gains, u , u_d are the horizontal coordinates of the actual and desired point in the image, v , v_d are the vertical coordinates of the actual and desired point in the image, and θ and θ_d are the actual and desired angles that the line makes with the positive v -axis in the image. Referring to Fig. 3, \dot{x}_v is the x component of the vehicle's velocity and is positive when the vehicle moves to the left. \dot{z}_v is the z component of the vehicle's velocity and is positive when the vehicle moves forward. $\dot{\psi}_v$ is the angular velocity of the vehicle and is positive when the vehicle rotates counter-clockwise. The subscript v is used to denote variables associated with the vehicle's frame. On the other hand, \dot{x} , \dot{z} , $\dot{\psi}$ are rates associated with the movement of features with respect to the camera frame. The relationship between these is discussed in Section II-C.

The motivation behind this controller is clear (see Fig. 3): If the line appears too far to the left in the image ($u > u_d$ — recall that u is positive to the left), then the robot should move to the left. Similarly, if the line appears too high in the image ($v > v_d$), the robot should move forward. Finally, if the line is rotated counter-clockwise beyond the desired angle in the image ($\theta > \theta_d$), then the vehicle should rotate counter-clockwise.

Consider the case

$$u_d = \theta_d = 0 \quad (6)$$

Thus, the desired image is a line which is centered horizontally and oriented vertically. Setting $u = \theta = 0$, $v = v_d$ in the projection map (1) gives desired values

$$x = 0, z = y\lambda/v_d, \psi = 0.$$

Thus, if the image line is in the desired position and orientation, the vehicle must be oriented parallel to the line ($\psi = 0$), and if the line were extended toward the vehicle, the vehicle would be located on this extended line ($x = 0$).

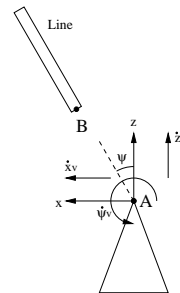


Fig. 3. Variables for Kinematic Equations. Point A should be understood as lying on an imaginary extension of the line. Note that the convention for ψ is consistent with Fig. 2.

There are several reasons why it is not clear if the controller will work. First of all, the relationship between image and actual positions is related by the perspective map, which is nonlinear. More significantly, however, the angle of the line in the image can sometimes direct the vehicle to rotate in the *wrong* direction. Recall that parallel lines in the world will map to converging lines in an image. Each of these has a different angle with respect to the positive v axis. Thus, if the vehicle takes corrective cues from the angle of an image line, it could easily rotate *away* from the correct orientation.

Subsequent sections will demonstrate the surprising result that the controller produces asymptotic convergence to the correct orientation and position for virtually any initial position and orientation of interest.

C. Vehicle Kinematic Equations

Fig. 3 illustrates the important variables involved in the vehicle kinematic equations. The equations can be derived from the standard relative motion equation for rigid bodies:

$$v_B = v_A + \omega \times r_{B/A},$$

where A and B are points on a rigid body, v_B is the velocity vector of B , v_A is the velocity vector of A , ω is the angular velocity vector of the body, and $r_{B/A}$ is the relative position vector drawn from A to B . Points A and B are labeled in Fig. 3, where point A is to be understood as lying on the line and not on the vehicle (it might be helpful to visualize an imaginary extension of the line). All calculations will be done with respect to the camera frame. $v_B = (\dot{x}, \dot{z})$ is the velocity of the point B as seen in the camera frame. As the vehicle moves with velocity (\dot{x}_v, \dot{z}_v) , the point A on the line (or its imaginary extension) will appear to move with velocity $-(\dot{x}_v, \dot{z}_v)$. Thus, $v_A = -(\dot{x}_v, \dot{z}_v)$. Clearly, $r_{B/A} = (x, z)$. Finally, the line will appear to rotate about the point A in the counter-clockwise direction if ψ is increasing. Thus, ω has magnitude $\dot{\psi}$ and is

in the counter-clockwise direction when $\dot{\psi} > 0$. Note that ψ increasing means $\dot{\psi}_v < 0$. Substituting these results into the relative motion equation and adding an equation corresponding to the previous statement gives the vehicle kinematic equations:

$$\dot{x} = -\dot{x}_v + \dot{\psi}z \quad (7)$$

$$\dot{z} = -\dot{z}_v - \dot{\psi}x \quad (8)$$

$$\dot{\psi} = -\dot{\psi}_v. \quad (9)$$

D. Final Model

Combining equations (1–9) gives the following set of differential equations:

$$\begin{aligned} \dot{x} &= -k_1\lambda\frac{x}{z} - k_3z\sin^{-1}\left(\frac{z\tan\psi - x}{\sqrt{(z\tan\psi - x)^2 + y^2}}\right) \\ \dot{z} &= -k_2\left(\lambda\frac{y}{z} - v_d\right) + \\ &\quad k_3x\sin^{-1}\left(\frac{z\tan\psi - x}{\sqrt{(z\tan\psi - x)^2 + y^2}}\right) \\ \dot{\psi} &= -k_3\sin^{-1}\left(\frac{z\tan\psi - x}{\sqrt{(z\tan\psi - x)^2 + y^2}}\right). \end{aligned}$$

Note that y can be considered a parameter since it is constant (From Fig. 2, it is the location of the ground with respect to the camera, which does not change). There appear to be 6 parameters: $k_1, k_2, k_3, \lambda, y, v_d$. However, some of these can be eliminated by using the following dimensionless variables and parameters:

$$\begin{aligned} \hat{x} &= \left(-\frac{1}{y}\right)x, \quad \hat{z} = \left(-\frac{1}{y}\right)z, \quad \hat{v}_d = \left(\frac{1}{\lambda}\right)v_d \\ \hat{t} &= \left(-\frac{\lambda k_1}{y}\right)t, \quad \hat{k}_2 = \left(-\frac{v_d}{\lambda}\frac{1}{k_1}\right)k_2, \quad \hat{k}_3 = \left(-\frac{y}{\lambda}\frac{1}{k_1}\right)k_3. \end{aligned}$$

Note that the expressions in parentheses are all constant and are all positive since $y < 0$ and $v_d < 0$. Solving for x, z, v_d, t, k_2, k_3 , and substituting into the original equations gives the dimensionless equations below. With the risk of introducing some confusion, the carets have been dropped for simplicity. Note that derivatives are with respect to dimensionless time.

$$\begin{aligned} \dot{x} &= -\frac{x}{z} - \\ &\quad k_3z\sin^{-1}\left(\frac{z\tan\psi - x}{\sqrt{(z\tan\psi - x)^2 + 1}}\right) \quad (10) \end{aligned}$$

$$\begin{aligned} \dot{z} &= -k_2\left(\frac{z + 1/v_d}{z}\right) + \\ &\quad k_3x\sin^{-1}\left(\frac{z\tan\psi - x}{\sqrt{(z\tan\psi - x)^2 + 1}}\right) \quad (11) \end{aligned}$$

$$\dot{\psi} = -k_3\sin^{-1}\left(\frac{z\tan\psi - x}{\sqrt{(z\tan\psi - x)^2 + 1}}\right). \quad (12)$$

Thus, through simple scaling, the system can be simplified to have only 3 independent parameters of interest: k_2, k_3, v_d . This new system will have all the same qualitative properties of the the original system. In particular, the stability properties of corresponding equilibria will be the same. Also, having obtained the system flow for the simplified system from a set of initial conditions, it is easy, in principle, to scale all variables and obtain the flow for the original system.

III. STABILITY ANALYSIS

The system in equations (10–12) has a single equilibrium at $(x, z, \psi) = (0, -1/v_d, 0)$ within the domain $D = \{(x, z, \psi) : z > 0, -\pi/2 < \psi < \pi/2\}$. Lyapunov's indirect and direct methods were used to test the stability of this equilibrium. Both demonstrated local asymptotic stability, but they provided different additional information.

A. Lyapunov's Indirect Method

Let $w = (x, z, \psi)$. The system in equations (10–12) is of the form $\dot{w} = f(w)$, with equilibrium $\bar{w} = (0, -1/v_d, 0)$. The linear approximation to the system at the equilibrium is given by

$$\frac{\partial f}{\partial w}(\bar{w}) = \begin{bmatrix} v_d - \frac{k_3}{v_d} & 0 & -\frac{k_3}{v_d^2} \\ 0 & k_2v_d & 0 \\ k_3 & 0 & \frac{k_3}{v_d} \end{bmatrix}.$$

The characteristic equation is then

$$(\lambda - k_2v_d)(\lambda^2 - v_d\lambda + k_3) = 0.$$

So, the eigenvalues are

$$\lambda = k_2v_d, \frac{v_d}{2} \pm \sqrt{\frac{v_d^2}{4} - k_3}.$$

All 3 eigenvalues have negative real parts, given the constraints $k_2 > 0, v_d < 0, k_3 > 0$. This leads to the following:

Theorem 1: If $k_2 > 0, k_3 > 0$, and $v_d < 0$, then the equilibrium $(x, z, \psi) = (0, -1/v_d, 0)$ of the visually-servoed, omni-directional robot equations (10–12) is locally asymptotically stable.

B. Lyapunov's Direct Method

Again, $w = (x, z, \psi)$, and equations (10–12) are of the form $\dot{w} = f(w)$, with equilibrium $\bar{w} = (0, -1/v_d, 0)$. Recall that $D = \{(x, z, \psi) : z > 0, -\pi/2 < \psi < \pi/2\}$. Consider the coordinate change $p = h(w), h : D \rightarrow \mathbb{R}^3, p = (q, r, s)$, given by

$$\begin{aligned} q &= \ln(-zv_d) \\ r &= z\tan\psi - x \\ s &= z\tan\psi + x \end{aligned}$$

The coordinate change is one-to-one, onto and has inverse $h^{-1} : \mathbb{R}^3 \rightarrow D$ given by

$$\begin{aligned} x &= \frac{s-r}{2} \\ z &= -\frac{1}{v_d} e^q \\ \psi &= -\tan^{-1} \left(\frac{v_d(s+r)e^{-q}}{2} \right) \end{aligned}$$

In order to conclude that D is a region of attraction for the system in equations (10–12), it will be necessary to consider the dynamics of the coordinate-transformed system. These dynamics are described by

$$\dot{p} = \left(\frac{\partial h^{-1}}{\partial p}(p) \right)^{-1} f(h^{-1}(p)) \quad (13)$$

and this system has equilibrium $p = 0$. This equilibrium will be shown to be globally asymptotically stable, and this will allow the conclusion that \bar{w} is an asymptotically-stable equilibrium for the original system, and D is a region of attraction.

Consider the Lyapunov function candidate for the original system

$$\begin{aligned} V(x, z, \psi) &= x^2 + \frac{(zv_d)^2 - 1 - 2\ln(-zv_d)}{v_d^2} - \\ &\quad \frac{2\ln(\cos \psi)}{v_d^2}. \end{aligned}$$

Composing this with h^{-1} gives

$$\begin{aligned} V(h^{-1}(q, r, s)) &= \frac{(s-r)^2}{4} + \frac{1}{v_d^2} (e^{2q} - 1 - 2q) - \\ &\quad \frac{2}{v_d^2} \ln \left(\frac{2}{\sqrt{4 + v_d^2(s+r)^2 e^{-2q}}} \right) \end{aligned}$$

It is clear that this is always positive except at $(q, r, s) = (0, 0, 0)$. Also, it can be shown that $\|p\| = \|(q, r, s)\| \rightarrow \infty$ means $V \circ h^{-1} \rightarrow \infty$. Due to space constraints, this is not demonstrated here.

Next, consider the derivative of the Lyapunov function candidate $V \circ h^{-1}$ along system trajectories of (13). With some algebra, it can be shown that following identity holds:

$$\frac{d}{dt} V(h^{-1}(p)) = \frac{\partial V}{\partial x}(h^{-1}(p)) f(h^{-1}(p))$$

Fortunately, this is simply $\dot{V} \circ h^{-1}$, which is much easier to compute than attempting to differentiate $V \circ h^{-1}$ along system trajectories of (13).

The derivative of V along system trajectories of the original system (10–12) is

$$\dot{V}(x, z, \psi) =$$

$$\begin{aligned} &-2\frac{x^2}{z} - 2k_2 \frac{(z + 1/v_d)^2 (z - 1/v_d)}{z^2} - \\ &2\frac{k_3}{v_d^2} z (z \tan \psi + x) \sin^{-1} \left(\frac{z \tan \psi - x}{\sqrt{(z \tan \psi - x)^2 + 1}} \right). \end{aligned}$$

Composing this with $h^{-1}(q, r, s)$ gives

$$\begin{aligned} \dot{V}(h^{-1}(q, r, s)) &= \\ &\frac{v_d e^{-q}}{2} (r-s)^2 + 2k_2 e^{-2q} \frac{(e^q - 1)^2 (e^q + 1)}{v_d} + \\ &2\frac{k_3 e^{-q}}{v_d} s \sin^{-1} \left(\frac{r}{\sqrt{r^2 + 1}} \right) \end{aligned}$$

Since $e^q > 0$ for all q , if it can be shown that $e^q (\dot{V} \circ h^{-1}) < 0$ in all of $\mathbb{R}^3 - (0, 0, 0)$, then, it will be possible to conclude that $\dot{V} \circ h^{-1} < 0$ in all of $\mathbb{R}^3 - (0, 0, 0)$. Multiplying by e^q gives

$$\begin{aligned} e^q (\dot{V} \circ h^{-1}) &= \\ &\frac{v_d}{2} (r-s)^2 + 2k_2 e^{-q} \frac{(e^q - 1)^2 (e^q + 1)}{v_d} + \\ &2\frac{k_3}{v_d} s \sin^{-1} \left(\frac{r}{\sqrt{r^2 + 1}} \right) \end{aligned} \quad (14)$$

Now, the second term on the right side of equation (14) is completely decoupled from the other terms, and it is negative if $q \neq 0$ (given the constraints $k_2 > 0$, $v_d < 0$). For this reason, the analysis will focus on the following function, which contains the remaining two terms:

$$g(r, s) = \frac{v_d}{2} (r-s)^2 + 2\frac{k_3}{v_d} s \sin^{-1} \left(\frac{r}{\sqrt{r^2 + 1}} \right).$$

First, rewrite g in the form

$$\begin{aligned} g(r, s) &= \frac{v_d}{2} (r^2 + s^2) - \\ &v_d \left(1 - 2\frac{k_3}{v_d^2} \frac{1}{r} \sin^{-1} \left(\frac{r}{\sqrt{r^2 + 1}} \right) \right) r s, \end{aligned}$$

and note the existence of the following limit:

$$\lim_{r \rightarrow 0} \frac{1}{r} \sin^{-1} \left(\frac{r}{\sqrt{r^2 + 1}} \right) = 1.$$

Now, compare g with the function

$$-\frac{1}{2}(r^2 + s^2) + ars,$$

which is less than zero for all $(r, s) \neq (0, 0)$ if $|a| < 1$. From this, it is clear that the condition to be satisfied is

$$-1 < 1 - 2\frac{k_3}{v_d^2} \frac{1}{r} \sin^{-1} \left(\frac{r}{\sqrt{r^2 + 1}} \right) < 1.$$

Since

$$0 < \frac{1}{r} \sin^{-1} \left(\frac{r}{\sqrt{r^2 + 1}} \right) < 1, \quad r \neq 0,$$

$g(r, s) < 0$ for all $(r, s) \neq (0, 0)$ if $k_3/v_d^2 < 1$. This means that $e^q(\dot{V} \circ h^{-1})$ in equation (14) is less than zero for all (q, r, s) in $\mathbb{R}^3 - (0, 0, 0)$. Finally, since $e^q > 0$ for all q , it is possible to conclude that $\dot{V} \circ h^{-1} < 0$ for all (q, r, s) in $\mathbb{R}^3 - (0, 0, 0)$.

At this point, all of the conditions for the Barbashin-Krasovskii theorem [14] have been satisfied for the system in the p coordinates (13) provided that $k_3/v_d^2 < 1$. Specifically,

$$V(h^{-1}(0)) = 0 \text{ and } V(h^{-1}(p)) > 0, \quad \forall p \neq 0,$$

$$\|p\| \rightarrow \infty \Rightarrow V(h^{-1}(p)) \rightarrow \infty, \text{ and}$$

$$\frac{d}{dt} V(h^{-1}(p)) < 0, \quad \forall p \neq 0$$

This means that for the coordinate-transformed system in (13), $p = 0$ is globally asymptotically stable. Given the nature of the coordinate transformation, it is possible to then conclude the following:

Theorem 2: If $k_2 > 0$, $k_3 > 0$, $v_d < 0$, and $k_3/v_d^2 < 1$, then $D = \{(x, z, \psi) : z > 0, -\pi/2 < \psi < \pi/2\}$ is a region of attraction for the asymptotically stable equilibrium $(0, -1/v_d, 0)$ of the visually-servoed, omnidirectional robot equations (10–12).

IV. SIMULATIONS

Simulations were performed to verify the analysis and to gain some understanding of the motion resulting from the simple proportional controller of Section II-B. Parameters for the simulation in equations (10–12) were $k_2 = 1$, $k_3 = 1$, $v_d = -1$. Fig. 4 shows x , z , and ψ (the coordinates of the line, as seen from the camera frame) as the system converges to the equilibrium, $(x, z, \psi) = (0, 1, 0)$. Note that ψ is initially at the correct value but (due to the angle determined from the image) initially increases as the controller attempts to drive the perceived error to zero. Also, note that the controller specifies an initial value for \dot{x}_v which has the correct sign (the vehicle moves left, as it should), but the rotation of the vehicle causes x , the x component of the line position as seen from the camera frame, to move in the wrong direction.

Fig. 5 shows the motion of the vehicle for the same simulation which generated the plot in Fig. 4. The vehicle is represented as a triangle, with the camera located at the narrow vertex. The vehicle begins by turning clockwise, since the initial image of the line is clockwise (See Fig. 6). It also moves forward and to the left. It moves forward because the starting point of the image line is above the desired value of $v = -1$.

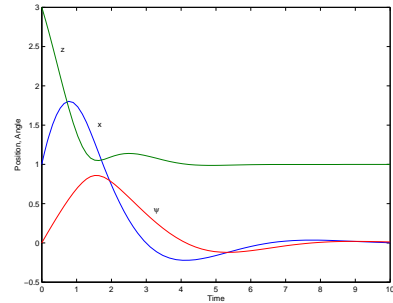


Fig. 4. Position and Location of Line in Camera Frame as Vehicle Moves.

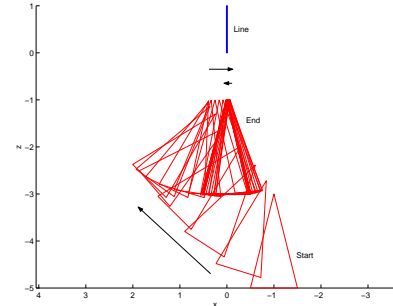


Fig. 5. Motion of Vehicle.

It moves to the left because the starting point of the image line is greater than the desired value of $u = 0$. Again, note that the vehicle initially had the correct orientation, but the simple controller of Section II-B caused it to move away from this orientation.

The frames in Fig. 6 show the motion of the image of the line in the camera, as the vehicle moves as indicated in Figure 5. Initially, the image line is at 45° clockwise with respect to the v -axis ($x = 1$, $\psi = 0$ in equation (2)). As the vehicle rotates to try to correct this perceived orientation error, the line is shifted more to the left in the image. The vehicle's motion to the left tends to counteract this effect, but the rotation causes more of a shift than the translation (see also the initial movement of x in the wrong direction in Fig. 4). As the vehicle moves forward, the image line moves down in the image. At an intermediate time, the image line is nearly vertical in the image, although the vehicle does not actually have the correct orientation. This occurs when the extension of the line passes through the y -axis of the camera frame (See fifth plot in Figs. 5, 6). When this occurs, $z \tan \psi = x$, and so $\theta = 0$ (equation (2)).

The motion of the vehicle is certainly not optimal. A major reason is the use of the image line angle (θ) to correct the vehicle angle (ψ). From equation (2) it is clear that θ is not a good indicator of ψ . The excessive rotation of the vehicle in Fig. 5 and the excessive

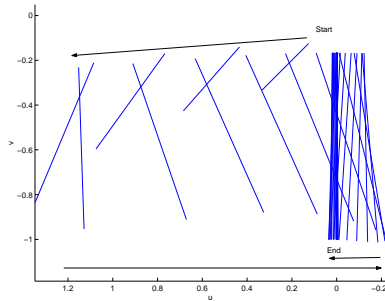


Fig. 6. Image of Line as Vehicle Moves.

motion of the line image in Fig. 6 can be reduced by using smaller values of the angle gain, k_3 .

V. EXTENSIONS

Recall that the model derivation assumed the vehicle rotated about the y -axis of the camera frame (Fig. 3). In actuality, the vehicle is currently programmed to rotate about its approximate center. A natural question is whether the simple proportional controller could still be used in this case, or whether the controller or vehicle program would need to be modified. To answer this, the kinematic equation (7) would need to be replaced by

$$\dot{x} = -\dot{x}_v + \dot{\psi}(z + L),$$

where L was the distance between the camera frame's y -axis and the vehicle's axis of rotation. A preliminary check using Lyapunov's indirect method showed that the equilibrium was locally asymptotically stable if $k_3L < -v_d$ (using dimensionless variables and parameters).

The vehicle is also currently programmed to accept changes in x , z , and ψ , rather than velocities. A natural question here is whether a linear discrete proportional controller would correctly align the vehicle if it replaced the controller in equations (3–5). This will likely be investigated in the near future.

VI. CONCLUSIONS

The majority of visual servoing work has made use of point features instead of lines. Also, for image-based servoing, it is common to calculate an image Jacobian matrix and use its inverse to determine corrections to robot position based on image errors. In this paper, lines in a plane were used as features, and although an image-based approach was used, no image Jacobian was needed. Despite this, asymptotic convergence to the desired location and orientation was proven for virtually any initial condition of interest. A simulation demonstrated successful servoing. The motion in this particular case included large, undesirable angle swings and image movement, but it was mentioned

that these could be reduced by decreasing the angle gain.

ACKNOWLEDGMENTS

The first author appreciates support from K. Moore (U.S. Army Automotive and Armaments Command Intelligent Mobility Program, agreement no. DAAE07-95-3-0023) and also assistance from S. Rich.

REFERENCES

- [1] K. Moore, N. Flann, Rich S., M. Frandsen, Y. Chung, J. Martin, M. Davidson, R. Maxfield, and C. Wood, "Implementation of an omni-directional robotic inspection system (ODIS)," in *Proceedings of SPIE Conference on Robotic and Semi-Robotic Ground Vehicle Technology*, Orlando, FL., May 2001.
- [2] Nicholas S. Flann, Morgan Davidson, Jason Martin, and Kevin L. Moore, "Intelligent behavior generation strategy for autonomous vehicles using a grammar-based approach," in *Proceedings of 3rd International Conference on Field and Service Robotics FSR2001*, Helsinki University of Technology, Otaniemi, Espoo, Finland, June 11 -13 2001.
- [3] H. G. Nguyen and J. P. Bott, "Robotics for law enforcement: beyond explosive ordnance disposal," Technical Report 1839, SPAWAR Systems Center, San Diego, SSC San Diego, San Diego, CA 92152-5001, Nov. 2000.
- [4] E. Poulson, J. Jacob, R. Gunderson, and B. Abbot, "Design of a robotic vehicle with self-contained intelligent wheels," in *Proceedings of SPIE Conference on Robotic and Semi-Robotic Ground Vehicle Technology*, vol. 3366, Orlando, FL, USA, 1998, pp. 68–73.
- [5] K. L. Moore and N. S. Flann, "A six-wheeled omnidirectional autonomous mobile robot," *IEEE Control Systems Magazine*, vol. 20, no. 6, pp. 53–66, 12 2000.
- [6] Seth Hutchinson, Gregory D. Hager, and Peter I. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, Oct. 1996.
- [7] Bernard Espiau, François Chaumette, and Patrick Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, June 1992.
- [8] Gregory D. Hager, "A modular system for robust positioning using feedback from stereo vision," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 4, pp. 582–595, Aug. 1997.
- [9] Nicolas Andreff, Bernard Espiau, and Radu Horaud, "Visual servoing from lines," in *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, San Francisco, CA, Apr. 2000, pp. 2070–2075.
- [10] Claus B. Madsen and Henrik I. Christensen, "A viewpoint planning strategy for determining true angles on polyhedral objects by camera alignment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, pp. 158–163, Feb. 1997.
- [11] José A. Castellanos and Juan D. Tardós, *Mobile Robot Localization and Map Building*, Kluwer Academic Press, 2000.
- [12] Darius Burschka and Gregory Hager, "Vision-based control of mobile robots," in *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*, Seoul, Korea, May 2001, pp. 1707–1713.
- [13] Lili Ma, Yangquan Chen, Matthew Berkemeier, Morgan Davidson, and Vikas Bahl, "Wireless visual servoing for an under car inspection mobile robot ODIS," in *Proceedings of the 15th IFAC World Congress on Automatic Control*, July 2002, submitted.
- [14] Hassan K. Khalil, *Nonlinear Systems*, Prentice-Hall, second edition, 1996.