

# Vision-based Control of a Smart Wheelchair for the Automated Transport and Retrieval System (ATRS)

Humberto Sermeno-Villalta and John Spletzer  
Department of Computer Science and Engineering  
Lehigh University  
Bethlehem, PA 18015

**Abstract**—In this paper, we present a vision-based control approach for autonomously docking a wheelchair onto a vehicle lift platform. This is a principle component of the Automated Transport and Retrieval System (ATRS) - an alternate mobility solution for drivers with lower body disabilities. The ATRS employs robotics, automation, and machine vision technologies, and can be integrated into a standard minivan or sport utility vehicle (SUV). At the core of the ATRS is a “smart” wheelchair system that autonomously navigates between the driver’s position and a powered lift at the rear of the vehicle - eliminating the need for an attendant.

From an automation perspective, autonomously docking the wheelchair onto the lift platform presented the most significant technical challenge for the proof-of-concept ATRS. This was driven primarily by geometry constraints, which limited clearance between the chair wheels and the lift platform rails. We present significant simulation and experimental results for our approach. These indicate that the coupling of vision-based localization for feedback and input/output feedback linearization techniques for controller design can provide for accurate wheelchair navigation and reliable docking under a range of ambient illumination conditions.

**Index Terms**—ATRS, smart wheelchair, vision-based control

## I. INTRODUCTION AND MOTIVATION

Automobile operators with lower body disabilities have limited options with regards to unattended personal mobility. The traditional solution is a custom van conversion that places the operator *in-wheelchair* behind the steering wheel of the vehicle. Entry/exit to the vehicle is also accomplished in-wheelchair via a ramp or powered lift device. Such a design has significant safety shortcomings. Wheelchairs do not possess similar levels of crash protection afforded by traditional motor vehicle seat systems, and the provisions used for securing them are often inadequate. However, regulating agencies often grant exemptions to vehicle safety requirements to facilitate mobility for disabled operators [1]. It should not be surprising then when research by the United States National Highway Traffic Safety Administration (NHTSA) showed that 35% of *all* wheelchair/automobile related deaths were the result of inadequate chair securement. Another 19% were associated with vehicle lift malfunctions [2].

The Automated Transport and Retrieval System (ATRS) is offered as an alternate mobility solution for operators with lower body disabilities. It employs robotics, automation, and machine vision technologies, and can be integrated into a standard minivan or sport utility vehicle

(SUV). At the core of the ATRS is a “smart” wheelchair system that autonomously navigates between the driver’s position and a powered lift at the rear of the vehicle. A primary benefit of this paradigm is the operator and chair are separated during vehicle operations as well as entry/exit. This eliminates the potential for injuries or deaths caused by both improper securement (as the operator is seated in a crash tested seat system) as well as lift malfunctions.

From an automation perspective, autonomously docking the wheelchair onto the lift platform presented the most significant technical challenge for the proof-of-concept ATRS. This was driven primarily by geometry constraints, which limited clearance between the chair wheels and the lift platform rails to ( $\pm 4$  cm). In response to this, we developed a vision-based solution for feedback control that can operate reliably within these specifications.

## II. RELATED WORK

Extensive work has been done in order to increase the safety levels of power wheelchairs while minimizing the level of human intervention. In these systems, the human operator is responsible for high-level decisions while the low-level control of the wheelchair is autonomous.

The Tin Man system [3], developed by the KISS Institute, automates some of the navigation and steering operations for indoor environments. The Wheelesley project [4], based on a Tin Man wheelchair, is designed for both indoor and outdoor environments. The chair is controlled through a graphical user interface that has successfully been integrated with an eye tracking system and with single switch scanning as input methods. The TAO Project [5] provides basic collision avoidance, navigation in an indoor corridor, and passage through narrow doorways. The system also provides landmark-based navigation that requires a topological map of the environment. The NavChair assistive wheelchair navigation system [6] uses feedback from ultrasonic sensors and offers obstacle avoidance, door passage, and wall following modes. More recently, the SmartChair [7], [8] uses a virtual interface displayed by an on-board projection system to implement a shared control framework that enables the user to interact with the wheelchair while it is performing an autonomous task.

A common theme in the above research is the robotics technology has been applied to assist or augment the skills of the chair operator. In contrast, the ATRS wheelchair

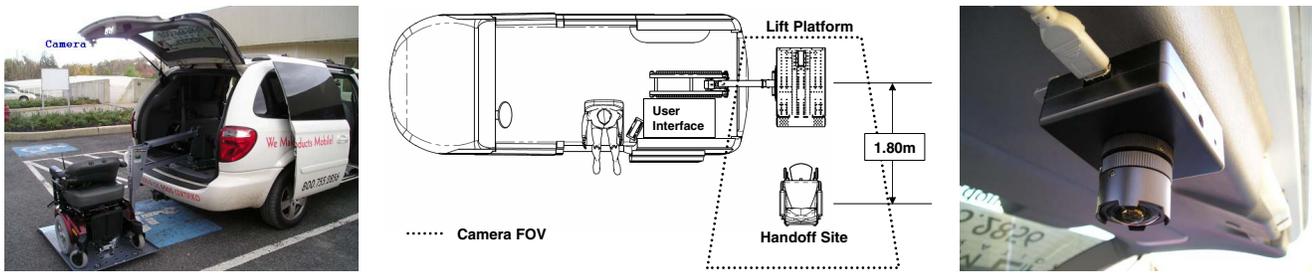


Fig. 1. Prototype ATRS system. (Left) The smart chair docked on the lift platform. (Center) Overhead layout of system components. (Right) The Dragonfly<sup>TM</sup> camera system used during development.

is in fact capable of autonomous vehicle navigation in outdoor environments. This can be realized because the operator is never seated in the chair during autonomous operations, and the chair always operates in the vicinity of the operator's vehicle. The former constraint mitigates operator safety issues, while the latter provides significant, invariant landmarks/features in an otherwise unstructured environment. What also makes the ATRS wheelchair attractive is that it can become commercially viable - providing a safer alternative to van conversions at roughly half the cost.

### III. SYSTEM OVERVIEW

The ATRS can be decomposed into five primary components: a smart wheelchair system, a vision system, a powered lift, a traversing power seat, and a user interface (UI) computer. From a robotics perspective, the smart wheelchair and vision system are the heart of the ATRS. These two subsystems allow the operator to be separated from the chair, and eliminate the need for an attendant. Figure 1 shows the prototype ATRS system and components.

To provide a better perspective on ATRS operation, we now summarize the operational procedures for a driver exiting his/her vehicle. The chair is initially located on a lift platform stowed in the cargo area of the vehicle. Once the lift platform is deployed by the operator, the role of the wheelchair is to shuttle from the lift platform to a position adjacent to the driver's seat to facilitate operator transfer. When the operator returns to the vehicle and transfers from the wheelchair to the driver's seat, the chair must be capable of navigating autonomously to the rear of the vehicle and reliably docking (locking in place) onto the lift platform. With the chair docked, the operator actuates the lift to return the platform and chair to the vehicle cargo area. When not operating autonomously, the ATRS wheelchair is placed in "manual mode," and operates no differently than any other powered wheelchair.

### IV. TECHNICAL APPROACH

The objective of this research was very specific: *Develop a means for reliable, autonomous docking (and undocking) of the ATRS wheelchair onto (and off of) the vehicle's lift platform.*

Any proposed solution was also required to live within system level constraints: alterations to the exterior of the

vehicle were prohibited, and (perhaps most significant from a computer vision perspective) sensor systems mounted within the van had to live within the UI's available CPU budget (1.1 GHz Pentium M).

The navigation task was decomposed into two phases. The first relied upon the LMS-200 and odometry to navigate the ATRS wheelchair from the driver's seat position to a designated handoff site at the rear of the vehicle. For the second phase, we integrated a camera into the vehicle's liftgate (Figure 1 (right)). Computer vision techniques were then used to provide state feedback to the chair over a dedicated RF link. Our motivation for employing a computer vision system was localization accuracy. Preliminary experiments with the LMS-200 indicated that position errors on the order of several centimeters were unacceptable for the docking task. We were able to achieve significantly better localization performance using a camera. However, the vision-based approach introduced its own challenges. The combination of the large footprint of potential wheelchair positions with the limited camera mounting height (1.83 meters) dictated a wide field-of view lens (for coverage) and high resolution CCD (for localization accuracy). These in turn introduced significant lens distortion and real-time processing challenges. Finally, the vision system would have to operate outdoors under a wide range of illumination levels. Addressing these challenges - and the subsequent control problem - is the focus of this paper.

### V. VISION-BASED TRACKING

To facilitate wheelchair detection and tracking, binary fiducials were affixed to each of the armrests. Furthermore, the following assumptions were made to simplify the tracking task include:

- 1) Wheelchair motion was constrained to the ground plane which is locally flat.
- 2) The position of the lift platform with respect to the van was fixed.
- 3) The position and orientation of the camera with respect to the lift platform was fixed.

With these assumptions, we reduced the problem of tracking the wheelchair to a two-dimensional pattern matching task. This was accomplished by composing geometric and distortion warps to a *virtual camera* frame directly overhead the expected fiducial positions. We now outline the details of the approach.

### A. Virtual Camera Image Warps

Our immediate problem was to infer the location of chair fiducials in the camera image. This is non-trivial due to significant perspective distortion from the rotation/translation of the camera frame. It is further aggravated by large radial and tangential distortion components from the wide field-of-view lens. However, recall from our assumptions that chair motion (and as a consequence fiducial position) is constrained to a single  $X$ - $Y$  plane. So, if we could place an orthographic camera overhead with its optical axis orthogonal to this plane, perspective distortion would be eliminated. A subsequent calibration for camera intrinsics would then eliminate lens distortions. This would preserve fiducial scale in the image, and reduce the problem to a plane search. Unfortunately, the camera pose is of course fixed - dictated by field-of-view requirements. As a result, we instead employ a virtual camera to serve this same purpose. This relies upon traditional image warping techniques, and is widely used in applications such as computer graphics and image morphing [9].

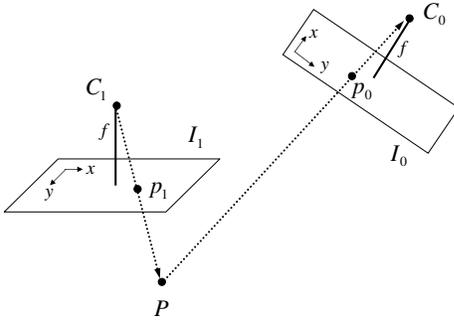


Fig. 2. Image coordinates from a virtual camera  $C_1$  are mapped back to the actual camera  $C_0$ . Corrections for both perspective and lens distortions can then be composed into a single warp from image  $I_0$  to  $I_1$ .

Let us assume that the position of a fiducial is estimated to be at position  $P_f = [X_f, Y_f, Z_f]^T$  in our world frame. The virtual camera is then moved to position  $C_1 = [X_f, Y_f, \bar{Z}]^T$  where  $\bar{Z}$  corresponds to a reference height from which the fiducial template was generated. We assume that our virtual camera is an ideal perspective camera with zero distortion and image center coincident with the optical axis. Other intrinsic parameters (e.g. focal length) are chosen the same as the actual camera. Referring to Figure 2, let  $p_1 = [x_1, y_1]^T \in \mathcal{I}_1$  denote the coordinates for a pixel in the virtual camera image  $\mathcal{I}_1$ . Our objective is to map  $p_1$  to a point  $p_0$  in the original camera image. To do this, we first project  $p_1$  to a world point  $P$  through our camera model

$$P = \begin{bmatrix} \frac{-(x_1 - cx_1)(Z_f - \bar{Z})}{f_x} + X_f \\ \frac{(y_1 - cy_1)(Z_f - \bar{Z})}{f_y} + Y_f \\ Z_f \end{bmatrix} \quad (1)$$

where  $(cx_1, cy_1)$  denote the image center coordinates of  $\mathcal{I}_1$ , and  $f_x, f_y$  corresponds to the equivalent focal length in pixels. Note that the  $Z$ -coordinate in the world will always

be set to  $Z_f$ . This is dictated by the ground plane constraint for the wheelchair. While this assumption will incorrectly map points that are not at the same height of the fiducials, it *will* map image points corresponding to the fiducials correctly.

Next we project  $P$  to camera coordinates in the true camera frame. Assuming that the camera has been well calibrated (for both intrinsic and extrinsic parameters), we obtain

$$P_C = \mathbf{R}_0^{-1}(P - C_0) \quad (2)$$

where the rotation matrix  $\mathbf{R}_0$  denotes the orientation and  $C_0 = [X_0, Y_0, Z_0]$  the position of the camera in the world frame. Prior to projecting  $P_C$  onto the image plane, we must also account for the significant image distortion admitted by the wide FOV lens. We first obtain the normalized image projection

$$p_n = [x_n, y_n]^T = [-X_C/Z_C, Y_C/Z_C]^T \quad (3)$$

from which we obtain the distorted projection as

$$p_d = (1 + k_1 r^2 + k_2 r^4) p_n + dP \quad (4)$$

where  $r = \sqrt{x_n^2 + y_n^2}$ ,  $k_1, k_2$  are radial distortion coefficients, and

$$dP = \begin{bmatrix} 2k_3 x_n y_n + k_4 (r^2 + 2x_n^2) \\ k_3 (r^2 + 2y_n^2) + 2k_4 x_n y_n \end{bmatrix} \quad (5)$$

is the tangential distortion component, with  $k_3, k_4$  the tangential distortion coefficients obtained from the intrinsic calibration phase. This distortion model is consistent with that outlined in [10], [11]. Finally, we can obtain the corresponding pixel coordinates in the true camera frame as

$$p_0 = \begin{bmatrix} f_x x_d + cx_0 \\ f_y y_d + cy_0 \end{bmatrix} \quad (6)$$

As  $p_0$  will not lie on exact pixel boundaries, we employed bilinear interpolation to determine the corresponding intensity value for  $p_1$  in the virtual image. The above process is then repeated for each pixel  $p \in \mathcal{I}_1$ .

### B. Fiducial Segmentation

1) *Image Processing*: To localize the wheelchair in the virtual camera image, we utilize normalized intensity distribution (NID) as a similarity metric [12]. An advantage of this formulation is that it explicitly models both changes in scene brightness and contrast from the reference template image. Additionally, in a comparison to alternate approaches in model based tracking, it was less sensitive to small motion errors [13].

Given a virtual image  $\mathcal{I}$ , an  $m \times n$  template  $T$  of the fiducial, and a block region  $B \in \mathcal{I}$  of equivalent size corresponding to a hypothetical fiducial position, the similarity of  $T$  and  $B$  can be defined as

$$\epsilon(T, \mathcal{I}, B) = \sum_{u=1}^m \sum_{v=1}^n \left[ \frac{T(u, v) - \mu_T}{\sigma_T} - \frac{B(u, v) - \mu_B}{\sigma_B} \right]^2 \quad (7)$$

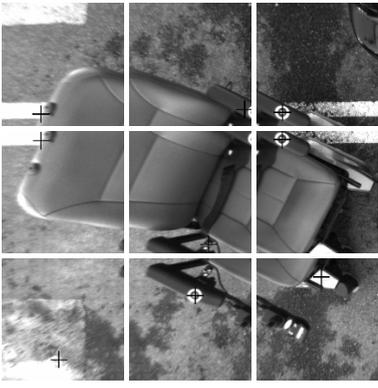


Fig. 3. Tessellation of the search window around the predefined handoff site. A black cross marks the 32x32 area with the highest NID value for each sub-window.

where  $T(u, v)$  denotes the grayscale value at location  $(u, v)$ ,  $\mu$  is the mean grayscale value, and  $\sigma$  the standard deviation. The image region

$$B^* = \arg \min(\epsilon) \quad \forall B \in \mathcal{I} \quad (8)$$

would then correspond to the fiducial position in the given image. One downside to correlation-based pattern recognition approaches is they are computationally expensive. To implement this in real-time, we first expand equation 7 and further simplify it for comparisons using a single template by eliminating the invariant factors. The resulting equation,

$$\epsilon_T(\mathcal{I}, B) = \frac{1}{\sigma_B} \left[ K \cdot \mu_B - \sum_{u=1}^m \sum_{v=1}^n T(u, v) \cdot B(u, v) \right] \quad (9)$$

where  $K = m \cdot n \cdot \mu_T$ , requires 1 division,  $m \cdot n$  additions, and  $m \cdot n + 1$  multiplications. We use fixed-point approximation with four bits for the fractional part to compute  $\sigma$  for all  $B \in \mathcal{I}$ , traversing  $\mathcal{I}$  from top to bottom, left to right. All calculations take advantage of the single-instruction, multiple-data (SIMD) instruction set available on the Pentium<sup>TM</sup> M family of processors.

### C. Localization

To compensate for the uncertainty in the initial position estimation, a search window  $W$  around a predetermined handoff position is tessellated into a set  $\mathcal{S}$  of overlapping subregions, as shown in figure 3. The number and dimensions of each subregion is determined by the search region size and the wheelchair dimensions. The diagonal distance in world coordinates of  $S_i \in \mathcal{S}$  is always less than the distance between the wheelchair arms, ensuring that at most one fiducial is completely visible in each subregion. The overlapping region dimensions are chosen such that  $\{B|B \in W\} = \bigcup_{i=1}^{|\mathcal{S}|} \{B|B \in S_i\}$ .

To identify the two wheelchair markers, NID is calculated for every member of  $\mathcal{S}$  using an  $m \times n$  template  $T$ . The set  $\mathcal{B}$  is formed from the best candidate blocks

$B_i^* \in S_i \quad \forall S_i \in \mathcal{S}$ . The pair of blocks  $P_{ij} = (B_i^*, B_j^*)$  is considered a valid wheelchair candidate if:

- The distance between  $B_i^*$  and  $B_j^*$  in world coordinates is within a predefined tolerance from the distance between the wheelchair fiducials.
- Their distance in pixel coordinates is greater than  $\max(m, n)$ .

Among the set  $\mathcal{P}$  of valid candidates, the pair of regions

$$P^* = \arg \max(\epsilon(T, S_i, B_i^*) + \epsilon(T, S_j, B_j^*)) \quad \forall P_{ij} \in \mathcal{P} \quad (10)$$

would then correspond to the fiducial positions in the given window. If no such pair exists, the size of the search window is expanded by the size of one tessellation subregion and the process is repeated. If the system fails to localize the wheelchair after two iterations of the above process, user intervention is required either to pinpoint its position on the camera image or to move it close to the handoff position and within the camera's field of view using a remote control interface.

Subsequent localization steps look for the fiducials in a  $144 \times 144$  pixel area centered around their last known positions. Even under our constrained CPU budget, localization could be accomplished at camera frame rate (15 Hz). This allowed for a maximum linear velocity of approximately 2.5 m/s. In practice, wheelchair velocity was limited to  $v(t) \leq 0.5$  m/s.

### D. Vision-based Feedback Control

With the vision system in place to provide state feedback to the wheelchair over a dedicated RF link, the task of our controller is to guide the chair onto the lift platform where it will lock into the docking station, thereby securing it from further movement. The wheelchair coordinate frame is centered on its drive axle. When the chair is ideally docked, its  $X_R$ - $Y_R$  axes are coincident with the  $X$ - $Y$  axes for the world frame.

While the motion problem might be classified as point-to-point, there is one caveat. The velocity  $v(t)$  of the chair at its objective pose  $[x, y, \theta]^T = [0, 0, 0]^T$  must be significantly greater than zero. This is a function of the docking procedure, which requires that a plough mounted to the chair bottom strike the dock with sufficient momentum to actuate the locking mechanism. As a result, we choose instead to treat it as a particular case of path following and apply I/O feedback linearization techniques to design a PD controller to drive the chair along the x-axis [14]. The kinematics in terms of the path variables become

$$\begin{aligned} \dot{x} &= v(t) \cos \theta \\ \dot{y} &= v(t) \sin \theta \\ \dot{\theta} &= \omega \end{aligned} \quad (11)$$

If we assume that  $v(t)$  is piecewise constant, we obtain

$$\ddot{y} = \omega v(t) \cos \theta \equiv u \quad (12)$$

and with error as  $e = y - y_d = y$  we obtain

$$u = -k_v \dot{y} - k_p y \quad (13)$$

which with Equations 11 and 12 yields

$$\omega = -k_v \tan \theta - \frac{k_p y}{v \cos \theta} \quad (14)$$

where  $\omega$  is the angular velocity command applied by the robot and  $k_v, k_p$  are positive controller gains. Thus, the controller design only requires observations of the chair state which are readily available from chair localization.

## VI. EXPERIMENTAL RESULTS

### A. Simulations

1) *Localization*: Objective evaluation of localization performance for the vision system is a non-trivial task. Nevertheless, we can estimate an upperbound on performance by evaluating effectiveness of the camera calibration. To do this, point correspondences to a set of known world points on a calibration grid were marked by hand on a test image. Reprojections of the same world points onto the image plane were also calculated from the calibration parameters. By assuming the respective points are coplanar, we can also estimate the potential accuracy of fiducial tracking. These results are summarized in Figure 4.

Reprojection errors averaged 1.0 pixels over the sample set. At the fiducial height, this corresponds to only several millimeters in position error. Again, these estimates are overly optimistic as they do not account for errors in fiducial segmentation, changes in camera calibration, *etc.* Still, they were repeatable even after (albeit limited) vehicle driving.

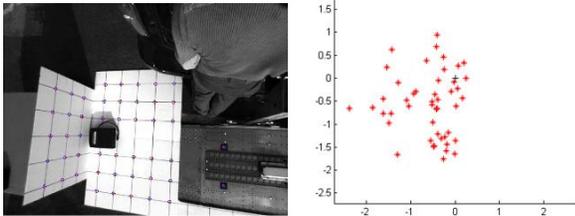


Fig. 4. Evaluation of calibration performance. Results indicate a potential tracking accuracy of several millimeters in position is possible

2) *Control*: Prior to operations on the chair itself, extensive simulations were conducted. In evaluating the PD version of the controller, we investigated its performance under critically damped behavior, *i.e.*,  $k_v = 2\sqrt{k_p}$ . For  $k_p$  values ranging from 0.1 to 2.0 (and corresponding values for  $k_v$ ), 25,000 simulated docking trials were run for each gain set. The initial position of the chair was constrained to a 1 meter square about the target handoff position - the specification for the system. Orientations were drawn at random from a zero mean Gaussian with  $\sigma = 0.1$  radians. Linear velocities were constant (0.3 m/s) and angular velocities were constrained to a maximum of 0.25 rad/s. Closed loop control was done at 15Hz. These values were identical to the actual system, and were dictated by safety considerations. To further enhance simulation fidelity, unmodelled zero-mean Gaussian noise

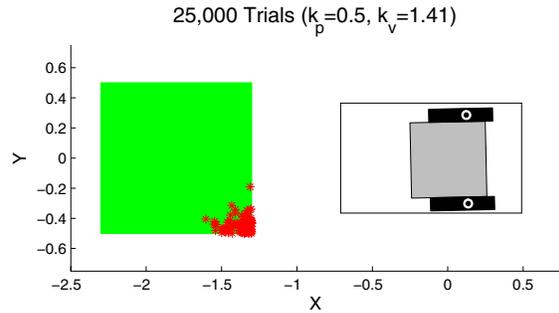


Fig. 5. Sample simulation results for a sub-optimal gain set. The failed initial poses (red stars) resulted from large orientation errors at a position near the lift platform.

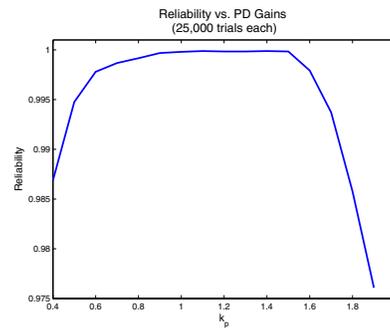


Fig. 6. Simulation results for critically damped behavior. Proportional gains in the range of 0.9 to 1.5 exhibited near perfect reliability under uncertainty.

was introduced into both the localization estimate as well as the control input to the vehicle.

Figure 5 shows the controller performance for the gain set  $k_p = 0.5, k_f = 1.41$ . In these simulations, a given trial was considered successful if the wheelchair position error  $e \leq 4\text{cm}$  and chair orientation  $|\theta| \leq 15^\circ$  at  $x=0$ . These values were based upon empirical observations of which final configurations would still permit proper docking. In this example, over 99% of trials were successful. Figure 6 shows controller performance across a range of gain sets. This indicates excellent performance for  $0.9 \leq k_p \leq 1.5$ . On the actual system, gains of  $(k_p, k_v) = (1.0, 2.0)$  yielded excellent performance.

### B. Experiments

Extensive experimentation was conducted with the ATRS system to support a public demonstration to industry representatives. This included 25 complete docking/undocking trials the day prior, as well as a day of demonstrations. Each of these trials was successful, with the vision-based controller demonstrating exceptional reliability and robustness to introduced errors. A representative docking trial is

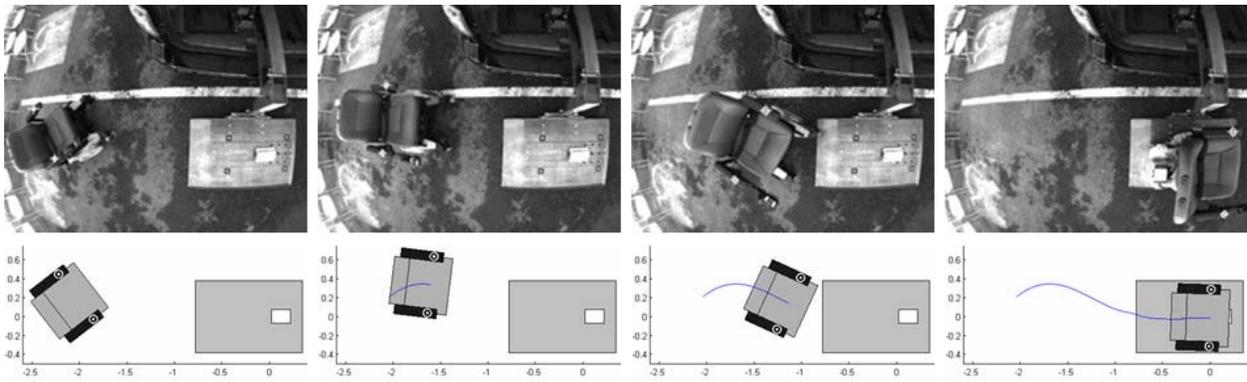


Fig. 7. Testing the vision-based control system with moderate position and orientation errors at the handoff site. The lower figure illustrates the path followed by the wheelchair as estimated by the vision system.

illustrated in Figure 7. This shows beginning, intermediate, and end poses of the chair as tracked by the vision system. The chair path as estimated by the vision system is also shown. A Dragonfly<sup>TM</sup> 1024x768 CCD camera with a 2.6mm focal length lens was used for all experiments.

## VII. CONCLUSIONS AND FUTURE WORK

The primary objective of this work was to determine the feasibility of using vision-based feedback control for autonomously docking a powered wheelchair onto a vehicle lift platform. Using a high resolution camera to provide state feedback, and I/O feedback linearization techniques for controller design, our approach was able to achieve a high level of docking reliability for the proof-of-concept ATRS under a broad range of illumination conditions.

We should however emphasize this was a proof-of-concept phase. There are several areas where the system needs to be significantly improved before commercialization can be realized. First and foremost, tracking robustness must be enhanced. While typically reliable, our fiducial tracking algorithms would fail under low light levels. Sunset generated a fairly consistent failure mode. This could potentially be rectified with additional lighting on the vehicle lift gate. Alternate detection/tracking algorithms (e.g. SIFT [15]) could also be considered.

Our controller design will also be upgraded to dynamically generate gains as a function of the settling distance required based upon the initial wheelchair pose.

Finally, our current approach assumes that both the intrinsic and extrinsic camera parameters remain fixed over time. The latter assumption may not be accurate. We are currently investigating a means by which a weak calibration is done at startup to estimate the extrinsic camera parameters. This can be facilitated by tracking features from the lift platform which are within the camera's field of view.

## ACKNOWLEDGMENTS

Special thanks to the entire ATRS Team, especially Herman Herman and Jeff McMahill (CMU) for all of their work on the wheelchair integration, Tom Panzarella, Jr. (Freedom Sciences), and Mike Martin and Tom Panzarella,

Sr. (Cook Technologies). Thanks to Rafael Fierro for discussions on I/O feedback linearization. Lastly, thanks to anonymous Reviewer 2 for the excellent feedback. This project was funded in part by a grant from the Commonwealth of Pennsylvania, Department of Community and Economic Development.

## REFERENCES

- [1] U.S. DOT NHTSA, *Vehicle Modifications to Accommodate People with Disabilities*, <http://www.nhtsa.dot.gov>.
- [2] U.S. DOT NHTSA, *Research Note: Wheelchair User Injuries and Deaths Associated with Motor Vehicle Related Incidents*, Sep 1997.
- [3] D. Miller and M. Slack, "Design and testing of a low-cost robotic wheelchair prototype," *Autonomous Robots*, vol. 1, no. 3, 1995.
- [4] H. Yanco, "Wheesley, a robotic wheelchair system: indoor navigation and user interface," *Lecture Notes in Artificial Intelligence: Assistive Technology and Artificial Intelligence*, pp. 256–268, 1998.
- [5] T. Gomi and A. Griffith, "Developing intelligent wheelchairs for the handicapped," *Lecture Notes in Artificial Intelligence: Assistive Technology and Artificial Intelligence*, pp. 150–178, 1998.
- [6] R. Simpson and S. Levine, "Automatic adaptation in the navchair assistive wheelchair navigation system," *IEEE Transactions on Rehabilitation Engineering*, vol. 7, no. 4, pp. 452–463, 1999.
- [7] S.P. Parikh, R.S. Rao, S. Jung, V. Kumar, J. Ostrowski, and C.J. Taylor, "Human robot interaction and usability studies for a smart wheelchair," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, Las Vegas, Nevada, 2003, pp. 3206–3211.
- [8] S.P. Parikh, V. Grassi, V. Kumar, and J. Okamoto, "Incorporating user inputs in motion planning for a smart wheelchair," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, New Orleans, LA, 2004, pp. 2043–2048.
- [9] S. Seitz and C. Dyer, "Image morphing," in *Proc. SIGGRAPH96*, 1996, pp. 21–30.
- [10] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Computer Vision and Pattern Recognition Conference*, 1997.
- [11] J-Y. Bouguet, "Camera calibration for matlab," Tech. Rep., Intel Corporation, 2001.
- [12] A. Fusiello, E. Trucco, T. Tommasini, and V. Roberto, "Improving feature tracking with robust statistics," *Pattern Analysis and Application*, vol. 2, pp. 312–320, 1999.
- [13] Ch. Grl, T. Ziner, and H. Niemann, *Pattern Recognition*, chapter Illumination insensitive template matching with hyperplanes, pp. 273–280, Springer, 2003.
- [14] A. Deluca, G. Oriolio, and C. Samson, *Robot Motion Planning and Control*, chapter Feedback control of a nonholonomic car-like robot, pp. 171–253, Springer-Verlag, 1998.
- [15] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60-2, pp. 91–110, 2004.