Proceedings of the
2005 IEEE Conference on Control Applications
Toronto, Canada, August 28-31, 2005

WC2.5

Vision-Based Hybrid Control Strategy for Autonomous Docking of a Mobile Robot

Dilan Amarasinghe, George K. Mann and Raymond G. Gosine
C-CORE and Faculty of Engineering and Applied Science
Memorial University of Newfoundland
St. John's, A1B 3X5, Canada
{*amarasin, gmann, rgosine*}*@engr.mun.ca*

***Abstract*— Parking or docking behavior is a very important behavioral unit for mobile robots. In a typical mobile robot application, robots have to navigate to a pre-defined destination and perform a given task. In order to perform those types of tasks, robots usually need to be precisely aligned against a target pose. In a vision-based mobile robot docking system, the nonholonomic nature and limited field–of–view constraints are required to be incorporated in the control strategy. In this paper we propose a novel hybrid control solution to this problem using image-plane measurements. The final docking position of the robot is defined as an image-plane reference. The robot is controlled to match the current view of the docking station to the reference view. The experimental results are used to demonstrate the effectiveness of the proposed approach. The results show that the proposed method provides a high level of robustness and precision which, is required in most practical applications.**

I. INTRODUCTION

The ability to align the robot against a predefined spatial configuration is a common requirement in most mobile robot applications. This alignment task can be interpreted in many ways depending on the operational context e.g. alignment against a door way may be part of a navigation task, alignment against a charging station represents a docking behavior. In this paper we collectively identify them as docking behaviors. The infrastructure required to support robot docking in a particular location is defined as a docking station. Docking behavior becomes a common requirement in mobile robotics due to two main reasons.

1) The level of precision provided by the localization techniques may not be adequate to execute a docking behavior after a series of navigational tasks.
2) The docking stations may change their position without notifying the robot.

Computer vision provides the unique ability to sense textural details of surfaces and is a convenient sensor for precise localization and alignment. Computer vision-based docking offers the ability to use simple and more general docking stations (simple visual patterns). Additionally, computer vision is a compact and low cost sensor popular in general robot control literature [1], [2].

This paper presents a novel solution to the docking problem of mobile robots. The proposed solution uses a simple image-based hybrid control strategy. The control strategy addresses the problems arising from nonholonomic nature of the robot and limited field–of–view constraints in order to achieve precise docking. The discrete behavior of the controller is implemented using a finite state machine and simple proportional control is used for continuous controllers. Finally, experimental results highlighting the effectiveness and the accuracy of the controller is presented.

A. Related Work

Problem of vision–based docking systems belong to the general research area of nonholonomic visual servoing. In visual servoing the two most popular methods are image–based (**IBVS**) and position–based visual servoing (**PBVS**) [1] where, error is directly calculated using image-plane measurements in the former while error is calculated in reference to a world coordinate system in the latter. In general, nonholonomic nature of a moving body impose velocity constraints on the rigid body. Further, when regular off the shelf cameras with limited field of view is used, they impose additional constraints on the control law. Thus, visual servoing of mobile robots is a particularly challenging task given the available number of degrees of freedom (usually two) and the limited field-of-view constraints of the camera.

Based on the intended applications, nonholonomic visual servoing can be categorized into path following techniques and docking techniques. Vision–based path following techniques [3], [4], [5], use features (stationary or moving) in the environment to continuously align itself. Continuous ground curves are the most commonly selected feature type for vision-based navigation [3]. In docking techniques the robot is aligned against a fixed set of features, such that the robot will satisfy a control objective that is defined in the image-plane. The involved control strategies can be generally categorized into conventional smooth controllers [6], [7] and switching [8], [9], [10], [11], [12] controllers.

Work in [6] provide a visual servoing technique based on target size and the horizontal position. Presented method is less robust in achieving a unique final robot pose for different starting positions. Thus, the method proposed in [6] has a limited applicability in docking applications where a unique final robot pose is expected. [7] provide an implementation of a conventional visual servoing scheme with and without a pan-able camera. Robot uses the turret in servoing without pan control adding an extra degree of freedom.

Hybrid (or switching) control has been a popular control technique in mobile robotics as it provides efficient method to overcome the nonholonomic constraints. Particularly, Lyapunov techniques have been widely adopted in closed-loop mobile robot control [13]. In a recent work [8], Murrieri *et al* presents a hybrid approach based on Lyapunov technique for vision-based robot docking. Further, the work provides solutions for both omnidirectional and limited field-of-view scenarios. The presented solution uses a PBVS approach which is known for its susceptibility to calibration and reconstruction errors. However, the presented method does not provide a way to utilize the field-of-view of the camera in the approaching stage resulting in a highly oscillatory behavior during parking. In [11] Hashimoto and Noritsugu proposes a visual servoing method for a nonholonomic mobile robot docking. The work is based on transforming robot dynamics into image-plane coordinate system. The solution only utilizes a small portion of the available field-of-view (i.e. the control law is confined to a small area of the image plane giving rise to highly oscillatory robot behavior). Kane *et al* [12] details a switching controller based on Lyapunov-like functions that uses a position based control technique with omnidirectional vision.

In other related works, Jose and Giulio [14] details two solutions based on optical flow for vision based docking. Docking station is comprised of an arbitrary pattern to facilitate the optical flow calculation. The method is only intended to achieve an approximate final orientation with respect to docking surface, thus it is less suitable for precision applications. In [15] Minten *et al* proposes a simpler solution based on color segmentation to dock daughter robots to a carrier.

B. Motivation

An image-based solution to the docking problem will greatly extend the flexibility of a mobile robot. A control law that only depend on the image-plane measurements (in pixels), will have the ability to provide image-plane templates (reference) for achieving required docking behavior. As the control strategy employs a finite state automaton, the system can be extended to facilitate many other servoing tasks such as integrated navigation systems [16]. Moreover when geometry of the target is known, metric space positioning can be readily achieved. This paper has the following contributions in the area of nonholonomic visual servoing of mobile robots. First, an accurate and robust vision-based robot docking system has been developed. Secondly, the field-of-view constraint is included into the control law using a finite state machine to obtain a guaranteed solution and to exploit the full coverage of the field-of-view. When the field-of-view is not fully utilized the system requires large number of switches between states to achieve the final goal.

Section II of the paper provides the preliminaries of the docking system and image-plane measurements. In section III the control strategy is presented. Section IV analyzes the convergence properties of the proposed system. Section V provides the details of the experimental implementation and the results.

II. Vision–Based Docking System

The objective of the docking system is to move the robot such that current view of the docking station accurately align with the reference docking station. To achieve this objective a hybrid controller based on finite state machine is proposed. In all image-plane measurements a regular pinhole camera model is assumed.

A. Docking Station

An equally spaced (horizontally aligned) three features define the docking station (Fig. 1). A minimum of three features are required to define a unique parking position. Image space docking configuration is considered to be at the center of the image (reference image). Docking station is assumed to be centered in the reference configuration. Although off center configurations can be used, a centered configuration robot will fully exploit the field-of-view constraints. Fig. 1 shows the docking station configuration. Measured quantities are the image-plane position of the three features. The values $[\ x_1,\ x_2,\ x_3,\ a_1,\ a_2,\ A\ ]$ (in pixels) can be calculated from the image space position of the three features. It should be noted that A, a_1 and a_2 are always positive quantities. Values of x_i have positive sign in the given direction. We define ΔA, as the difference between actual spacing (A) between the two outer features and the required (reference) spacing (A^*),

$$\Delta A = A - A^*.$$

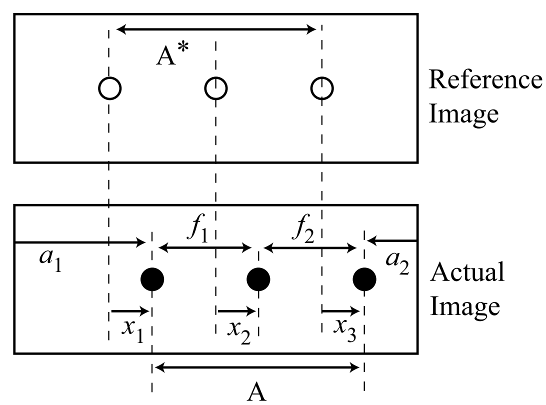


Fig. 1. Refrence and actual image of the docking station

Robot is considered docked (`parked`), when the following condition is satisfied

$$(|\Delta A| < \delta_A)^\wedge(|x_1| < \delta_x)^\wedge(|x_2| < \delta_x)^\wedge(|x_3| < \delta_x). \quad (1)$$

Where δ_A and δ_x are threshold values (in pixels) used to define the accuracy of the final image feature positions with respect to the reference feature positions.

This docking condition corresponds to a region around the desired metric robot pose. The area of this region (accuracy of docking) can be adjusted using δ_x and δ_A, to suit the application.

B. Overall Control System

Fig. 2 shows the overall block diagram of the closed loop control system. Hybrid controller uses the image plane measurements resulting from image processing and the image plane error to produce the control command for the robot (heading velocity and rotational velocity).

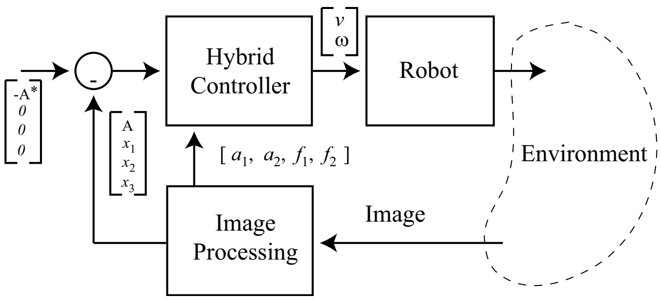


Fig. 2. Overall block diagram of the control system.

C. Assumptions

1) Robot operates on a flat floor.
2) Features are at approximately the same height as the horizontal optical axis of the camera.
3) Camera is fixed and optical axis is approximately parallel to the heading velocity.
4) At the start, the docking station is within the field-of-view of the camera.

III. Control Startegy

The control strategy proposed in this section is mainly motivated by behavior-based robotics [16]. Behavior–based robotics provide many biologically inspired intelligent control techniques for mobile robot navigation. Following the principles of behavior-based robotics, the proposed method provides close coupling between sensory information and motor control using simple mathematical relationships. Specifically, a finite state machine is used at the heart of the docking control system to provide the context (state) of operation (relationship between sensory information and motor speed) based on the current sensory information and the progress of the docking process.

A. Finite State Machine

The finite state machine (**FSM**) has four states and six transitions. Three states represent active controllers while the other two are initialization and termination states. Fig. 3 shows the FSM designed to solve the docking problem. k_i's are suitably chosen parameters. When the robot enters a state, the actions are described under the following three categories. (1) init: action when the robot enters state (one time), (2) during: actions that robot will execute while in this state (loop) and (3) exit: action upon leaving the state (one time). Further, robot will evaluate switching conditions from a state while in the state (during).

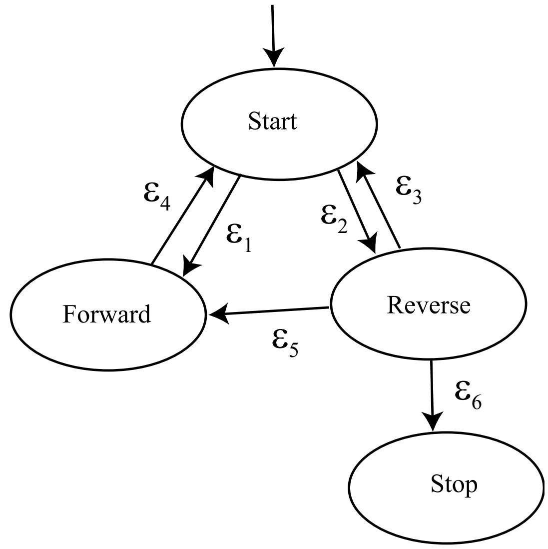


Fig. 3. Proposed finite state machine

1) *Start state*: In the start state, the robot will be oriented such that the feature in the center of the docking station aligns to the center of the image. Then it will switch the control of the robot to either 'forward' or 'reverse' state depending on the relative size of the docking station in the current image.

Algorithm 1 Start state

init: none
during:
while $|x_2| < \delta x$ **do**
  $v = 0$
  $\omega = -0.5 k_1 x_2$
end while
if $\Delta A < 0$ **then**
  exit event $= \epsilon_1$
else
  exit event $= \epsilon_2$
end if
exit:
$f_1 = x_2 - x_1$
$f_2 = x_3 - x_2$

2) *Forward*: When the robot is in this state, depending on the values of f_1 and f_2, it will align a side feature close to the corresponding edge of the image while moving towards the docking station. The robot will exit this state when the docking station threatens to leave the field-of-view.
3) *Reverse*: During the reverse state the robot will move away from the docking station while keeping the center feature in the middle of the image frame. The reverse state is necessary to move the docking station sufficiently inside the image to facilitate the next forward maneuver. Additionally, while the robot is in the reverse state, it will monitor the docking condition

Algorithm 2 Forward state

init: none
during:
while $a_1 < a_{limit}$ **OR** $a_2 < a_{limit}$ **do**
  $v = v_c$
  if $f_1 > f_2$ **then**
    $\omega = k_2 x_1$
  else
    $\omega = -k_2 x_3)$
  end if
end while
exit event $= \epsilon_1$
exit: none

(`parked`, (1)).

Algorithm 3 Reverse state

init: none
during:
while $dA < A_{min}$ **OR** (**NOT** `parked`) **do**
  $v = -v_r$
  $\omega = -k_1 x_2$
end while
if $dA < A_{min}$ **then**
  exit event $= \epsilon_5$
else if `parked` **then**
  exit event $= \epsilon_6$
end if
exit: none

4) *Stop*: Robot will come to a halt after a successful docking at stop state.

Algorithm 4 Stop state

init: none
during:
$v = 0$
$\omega = 0$
exit event = none
exit: none

IV. Analysis of Covergence

Fig. 4 shows a typical robot orientation at the exit of the 'start' state. It can be easily shown that $\beta_1 = \beta_2$ when $x = -c$ i.e. robot should park at some where on the dotted line in Fig. 4 facing the docking station. In order to illustrate the convergence properties of the proposed method, a contour plot of $|f_1 - f_2|$ for different robot positions is used. From Fig. 4,

$$|f_1 - f_2| = \lambda |\tan(\beta_1) - \tan(\beta_2)|$$

where λ is the focal length of the camera. Therefore,

$$|f_1 - f_2| \ \propto \ |\tan(\beta_1) - \tan(\beta_2)|.$$

Thus, for a qualitative discussion we can use the contour plot of constant $|\tan(\beta_1) - \tan(\beta_2)|$ for different robot positions in front of the docking station.

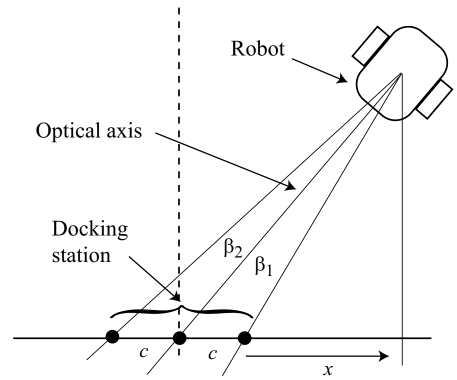


Fig. 4. Robot position and orientation at the exit of the 'start' state

Fig. 5 shows a contour plot of constant $|\tan(\beta_1) - \tan(\beta_2)|$ and a typical robot path. In order for the robot to converge to the final docking position the contour values must decrease. With the given behavior in the 'reverse' state, the robot always travels from high values to low values. At the end of each 'forward' travel, the robot will end up being in a lower value position in the contour than at the end of the previous 'forward' travel. Thus, by switching between the 'forward' and 'reverse' behaviors, docking process is guaranteed to converge.

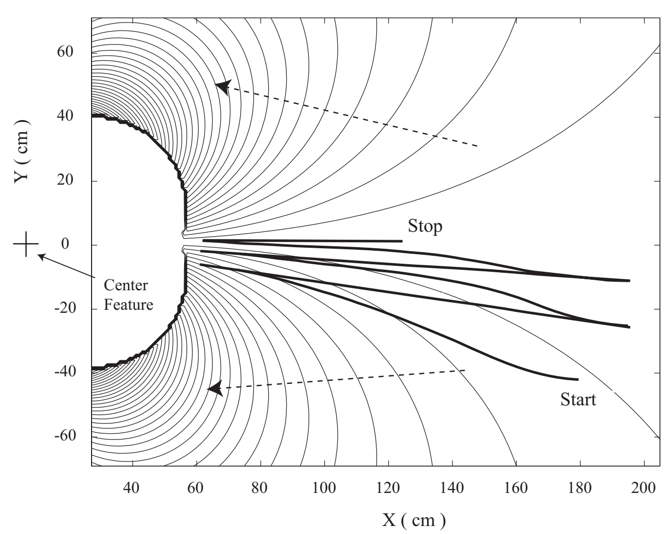


Fig. 5. Contour plot of constant $|\tan(\beta_1) - \tan(\beta_2)|$ for different robot positions in front of the docking station. Arrows indicate the direction of increasing value in contours. The semi-circle without countours corespond to the area where field-of-view constraints are violated.

V. Experiment and Results

The proposed method was implemented using Pioneer 3AT mobile robot. Fig. 6 shows the robot and the docking station. A web camera was used to acquire the images from the robot. Image processing routines and the finite state machine were implemented on a piggyback laptop computer. Image acquisition, processing and analysis is carried out at 30 frames per second. TABLE I lists the parameters used for the results described below.

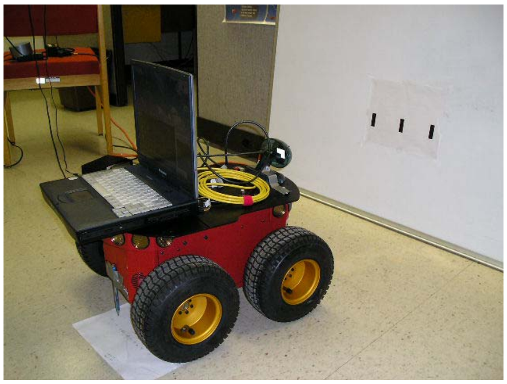


Fig. 6. Pioneer 3AT robot and the docking station

Parameter	Value
k_1	0.2
k_2	0.05
a_{limit}	60 pixels
c	106 mm
A^*	300 pixels
Feature 1 ref. position	170 pixels
Feature 2 ref. position	320 pixels
Feature 3 ref. position	470 pixels
v_c	40 mm / sec
v_r	60 mm / sec
δ_x	2 pixels
δ_A	2 pixels

TABLE I
SELECTED PARAMETER VALUES IN DOCKING SYSTEM

In the results shown here the robot starts from the left side of the docking station. As shown in Fig. 7, after three maneuvers, the desired image-plane positions for the features are achieved. It can be also seen that when the robot moves from the left side ('forward' state), left most feature (bottom line in Fig. 7) is maintained close to the a_{limit} value from the edge while the other corner feature approaches the other edge (top line in Fig. 7). Furthermore, Fig. 7 shows that during the 'reverse' state, the middle feature is always kept at the center of the image.

From the heading velocity curve (Fig. 8) it is apparent that when the robot reverses, the speed drops to half the nominal reverse velocity at the middle of the reverse maneuver. If the robot maintains the same high velocity in the docking region, there is a possibility of overshooting the final docking position as robot might be in the process of acquiring an image. This is a significant design consideration when low

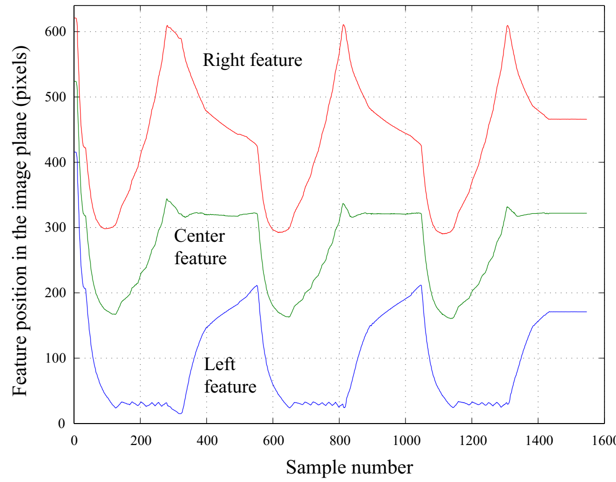
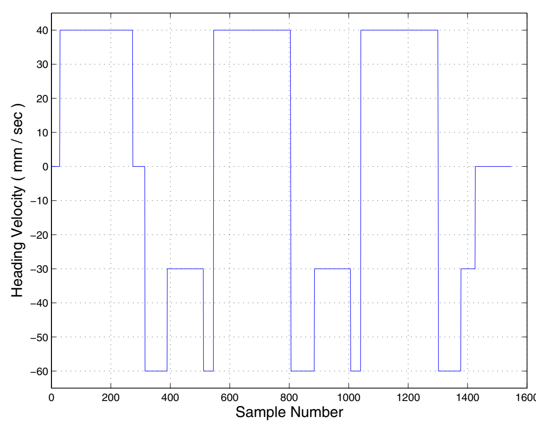
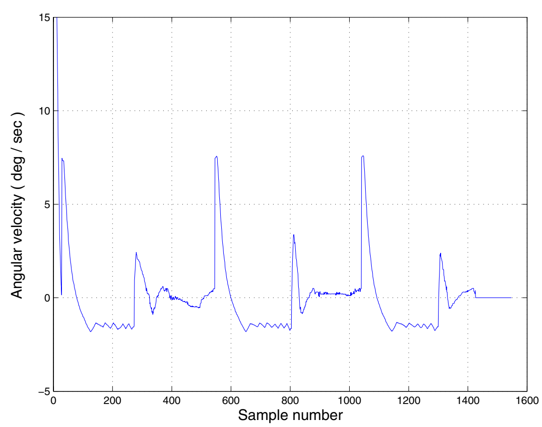


Fig. 7. Image-plane position of the three features. Image width of 640 pixels is represented by the vertical axis. The final (required) feature positions are 170, 320 and 470 pixels.



(a)



(b)

Fig. 8. Robot velocity during docking maneuvers. (a) Heading velocity (mm / sec) and (b) angular velocity (deg / sec)

frame rate image processing is used. In this implementation a discrete velocity change is selected when $|\Delta A| < 50$. Another possibility is to provide smooth reverse behavior based on the ΔA value as opposed to a discrete case.

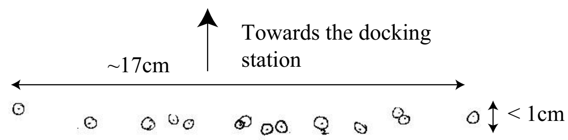


Fig. 9. Final parking positions for 15 different arbitrary starting positions

Fig. 9 shows the final docking positions for 15 different arbitrary starting positions with $\delta_x = \delta_A = 2$ pixels. The robot achieves less than 1 cm accuracy in the x-direction, accuracy in the y-direction is about 17 cm. It should be noted that for detecting the rectangular features a simple thresholding method followed by a blob analysis is used. The lower precision in y-direction can be mainly attributed to the repeatability of the image-plane measurements with 2 pixel accuracy.

A. Limitations of the method

The accuracy of the docking process can be improved by lowering the values of δ_A and δ_x. One observed shortcoming is that with lower δ_A and δ_x thresholds, the proposed method might show oscillatory behavior around the final docking position. Another inherent limitation is that, if lens or lens/imaging sensor combination is asymmetrical it will be impossible to achieve the final docking condition.

VI. Conclusion and Future Work

This paper has described a novel image-based method to achieve docking behavior for nonholonomic mobile robots. The reference docking configuration is described in the image-plane and is used by the robot to align itself to achieve the desired final pose. In this work we use a configuration based on 3 image features. Image-plane positions of the three features are used to control the robot during the whole duration of the docking maneuver. Results from the experiment show that this method has very high potential to be used as a practical docking system, with centimeter accuracy. Using a better camera (less motion blur and noise) and with improved lighting, the accuracy of the docking can be adjusted using the system parameters. Compared to other vision-based docking systems [8], [11] the proposed method utilizes full field-of-view (Fig. 7) during the entire docking process. This contributes to the increased efficiency (lesser number of maneuvers) in the docking process.

Future work on this method include developing algorithms to minimize the number of required maneuvers (thus reducing the docking time). Specially by incorporating more intelligence in the 'reverse' state, docking time can be further reduced. Described method also has the potential to be used in vision based navigation systems when a navigation task can be decomposed into a series of alignment (or parking tasks). With efficient algorithms to reduce the alignment time, such navigation a system will be able to contribute to a new method for vision based robot control.

Acknowledgment

The authors would like to thank Natural Sciences and Engineering Research Council of Canada (NSERC), C-CORE and Memorial University of Newfoundland for providing financial assistance for the research described above. The equipment grant by Canadian Foundation for Innovation (CFI) is gratefully acknowledged.

References

[1] Seth Hutchinson, Gregory D Hager, and Peter I Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, 1996.

[2] Lee E. Weiss, Authur C. Anderson, and Charles P. Neuman. Dynamic sensor–based control of robots with visual feedback. *IEEE Journal of Robotics and Automation*, 3(5):404–416, 1987.

[3] Yi Ma, Jana Kosecka, and Shankar S. Sastry. Vision guided navigation for a nonholonomic mobile robot. *IEEE Transactions on Robotics and Automation*, 15(3):521–536, June 1999.

[4] Rene Vidal, Omid Shakernia, and Shankar Sastry. Formation control of nonholonomic mobile robots with omnidirectional visual servoing and motion segmentation. In *Proceedings of the International Conference on Robotics and Automation*, pages 584–589, September 2003.

[5] D. Khadraoui, C. Debain, R. Rouveure, P. Martinet, P. Bonton, and J. Grace. Vision–based control in driving assistance of agricultural vehicles. *International Journal of Robotics Research*, 17(10):1040–1054, October 1999.

[6] Hong Zhang and James O. Ostrowski. Visual motion planning for mobile robots. *IEEE Transactions on Robotics and Automation*, 18(2):199–208, April 2002.

[7] Jose R. H. Carvalho, Patrick Rives, Aliton Santa-Barbara, and Samuel S. Bueno. Visual servo control for a class of mobile robot. In *Proceedings of the International Conference on Control Applications*, pages 431–436, September 2000.

[8] Pierpaolo Murrieri, Daniele Fontanelli, and Antonio Bicchi. A hybrid-control approach to parking problem of a wheeled vehicle using limited view–angle visual feedback. *The International Journal of Robotics Research*, 23(4–5):437–448, April-May 2004.

[9] Fabio Conticelli, Benedetto Allotta, and Pradeep K. Khosla. Image-based visual servoing of nonholonomic mobile robots. In *Proceedings of the International Conference on Decision and Control*, pages 3496–3501, December 1999.

[10] F. Conticelli, D. Prattichizzo, F. Guidi, and A. Bicchi. Vision-based dynamic estimation and set-point stabilization of nonholonomic vehicles. In *Proceedings of the International Conference on Robotics and Automation*, pages 2771–2776, 2000.

[11] Koichi Hashimoto and Toshiro Noritsugu. Visual servoing of nonholonomic cart. In *Proceedings of the International Conference on Robotics and Automation*, pages 1719–1724, April 1997.

[12] Kane Usher, Peter Ridley, and Peter Corke. Visual servoing of a car-like vehicle – an application of omnidirectional vision. In *Proceedings of the International Conference on Robotics and Automation*, pages 4288–4293, September 2003.

[13] M. Aicardi, G. Casalino, A. Bicchi, and A. Balestrino. Closed loop steering of unicycle–like vehicles via lyapunov techniques. *IEEE Robotics and Automation Magazine*, pages 27–35, March 1995.

[14] Jose Santos-Victor and Giulio Sandini. Visual behaviors for docking. *Computer Vision and Image Understanding*, 67(3):223–238, September 1997.

[15] Brian W. Minten, Robin R. Murphy, Jeff Hyams, and Mark Micire. Low order complexity vision based docking. *IEEE Transactions on Robotics and Automation*, 17(6):922–930, December 2001.

[16] Ronald C. Arkin. *Behavior Based Robotics*. MIT Press, Cambridge, MA, 1998.