2012 IEEE International Conference on Robotics and Automation
RiverCentre, Saint Paul, Minnesota, USA
May 14-18, 2012

Visual Servoing Control of a 9-DoF WMRA to Perform ADL Tasks

William G. Pence, Fabian Farelo, Redwan Alqasemi, Yu Sun, and Rajiv Dubey

Abstract—**The wheelchair-mounted robotic arm (WMRA) is a mobile manipulator that consists of a 7-DoF robotic arm and a 2-DoF power wheelchair platform. Previous works combined mobility and manipulation control using weighted optimization for dual-trajectory tracking [7]. In this work, we present an image-based visual servoing (IBVS) approach with scale-invariant feature transform (SIFT) using an eye-in-hand monocular camera for combined control of mobility and manipulation for the 9-DoF WMRA system to execute activities of daily living (ADL) autonomously. We also present results of the physical implementation with a simple "Go to and Pick Up" task and the "Go to and Open the Door" task previously published in simulation, using IBVS to aid the task performance.**

I. Introduction

THE 2010 US Census Bureau report on disability shows that about 10 percent of the working age population has some sort of disability, with most of these disabilities being ambulatory [1]. It has been shown that robotic arms can serve as effective assistive devices for users with impaired upper-body functions [2]. Two prototype WMRAs have been developed at the Center for Assistive, Rehabilitation and Robotics Technologies at the University of South Florida that outperform traditional 6-DoF WMRAs that are commercially available [3, 4].

Several user interfaces have been implemented on the 9-DoF WMRA system such as the 3D SpaceBall joystick, laptop touch screen, voice recognition, eye gaze tracking [6], and P300 brain-computer interface (BCI) [5]. Although the system has executed several ADL tasks successfully, it is difficult to teleoperate the 9-DoF system with combined mobility and manipulation.

In [7], dual-trajectory control was implemented to provide sub-trajectories in order to execute a "Go to and Open the Door" task in simulation. In this work, we theorize a visual servoing technique to control combined mobility and manipulation on the 9-DoF WMRA system for execution of ADL tasks. We use a simple 2D visual servoing method while approaching the goal object such that it will be in the workspace of the manipulator, then after a threshold distance from the goal object we use IBVS with SIFT [8], where the manipulator translates and orients to grasp the goal object and depth is estimated with a proximity sensor. Weighted optimization is used throughout the control system to control mobility and manipulation simultaneously in a coordinated manner. Finally, physical results of "Go to and Pick Up" and "Go to and Open the Door" ADL tasks are presented. This work presents a novel application of visual servoing to a combined 9-DoF mobile manipulator for the execution of assistive ADL tasks, and also presents a hardware implementation of previously published work in simulation [7], using IBVS with SIFT to enhance task execution robustness.

II. Background

Redundant mobile manipulators have become increasingly popular in the research field and can be used in various different applications. Having a mobile platform greatly increases the workspace of a manipulator since the system is able to navigate around in the environment. Control systems for a 7-DoF mobile manipulator consisting of a 5-DoF robotic arm mounted to a nonholonomic 2-DoF mobile platform were described in [9]. In this work, control of mobility and manipulation were decoupled such that the mobile platform moved to an area that put the goal object in the workspace of the manipulator, and then the manipulator grasped the goal object. In [10], combined kinematics for a nonholonomic platform with a manipulator were presented. In this work, redundancy in the system was resolved using the projected gradient and reduced gradient optimization methods. A sample trajectory was followed where the manipulator stayed in a pre-specified orientation while the mobile platform followed a circle.

Weighted least norm solution is one method of redundancy resolution, described in [11]. This method resolves redundancies in redundant manipulators while at the same time minimizing unnecessary motion of the joints. This can also be extended to avoid joint limits by using specific criterion functions described in [12]. In a similar fashion, we will coordinate mobility and manipulation using weighted optimization in this work.

Visual servo control has become increasingly popular due to its simplicity and robustness, especially for physical applications in real-world environments. These methods have been described in great detail in tutorials such as [13, 14]. Two main forms of visual servoing exist, but we mainly focus on IBVS where velocity control for the system is computed based strictly on features immediately available in

Manuscript received September 15, 2011. This work was supported in part by the National Science Foundation (NSF) under Grant IIS-0713560.

W. G. Pence is with the University of South Florida, Tampa, FL 33620 USA (phone: 813-974-2115; fax: 813-974-3539; e-mail: wpence@mail.usf.edu).

F. Farelo is with the University of South Florida, Tampa, FL 33620 USA. (e-mail: ffarelo@mail.usf.edu).

R. Alqasemi is with the University of South Florida, Tampa, FL 33620 USA. (e-mail: alqasemi@usf.edu).

Y. Sun is with the University of South Florida, Tampa, FL 33620 USA. (e-mail: yusun@cse.usf.edu).

R. Dubey is with the University of South Florida, Tampa, FL 33620 USA. (e-mail: dubey@usf.edu).

978-1-4673-1404-6/12/$31.00 ©2012 IEEE

the image. Position-based visual servoing (PBVS) is another form that relies on estimation of 3D positions in the image. Essentially, visual servoing provides a correspondence between matched features in a goal and camera image and robot movement, and gives as output a velocity controller for a robot system. The goal image contains a view of the goal object from the desired grasping position and orientation. Visual servoing minimizes the error between this goal image and camera image until it becomes zero, such that the goal object can then be grasped. This error is computed by matching features in the goal and camera image, and virtually any feature matching algorithm can be used.

Several works have demonstrated the success of visual servoing on fixed-base manipulators. In [15], the commercially-available Manus arm was controlled using a visual servoing technique relying on color-based feature extraction. This implementation was reliable, but depended on good color information for goal objects to be grasped. An extension to this work described in [16] used SIFT to match features between the goal and camera image. This provided a much more robust system working towards autonomous grasping. In [17], a similar approach was used by implementing a 2 1/2D homography-based visual servoing using SIFT. This work split the grasping task into gross and fine motion with separate control systems for each phase. While this work did not implement a full 3D visual servoing technique, it did provide a beginning to end solution for autonomous grasping.

Visual servoing applications on mobile manipulators have typically consisted of very simple manipulators on two-wheeled platforms. The 7-DoF mobile manipulator mentioned above was controlled using visual servoing in [18]. This work integrated IBVS and Q-learning to control the nonholonomic 2-DoF platform and 5-DoF manipulator. Control was decoupled such that the mobile platform first approaches the goal object, and then the manipulator grasps it. One of the problems with the application of visual servoing to mobile manipulators is that the mobile platform can easily move such that the visual features move outside the view of the camera and are therefore lost. Q-learning is used to aid the system in [18] to overcome this problem.

Although these applications to both fixed-base and mobile manipulators prove that the implementation of visual servoing control can be very successful, they still have some shortcomings. Fixed-base manipulators are constrained to their local workspace, which is undesirable for assisting users with daily ADLs. Applications of visual servoing to mobile manipulators typically decouple the control systems. In this work, we use a combined approach based on weighted optimization to integrate IBVS for full 3D control using SIFT on the 9-DoF WMRA system so that all DoF are controlled simultaneously throughout the entire task. This provides for a smoother and more seamless application specifically for the execution of ADL tasks.

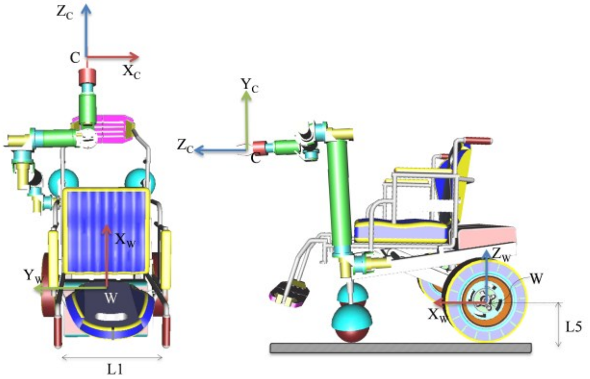

Fig. 1. WMRA coordinate frames. W shows the wheelchair frame and C shows the camera frame.

III. Control Systems

A. WMRA Combined Kinematics

For the 7-DoF manipulator, numerical solutions exist to have it follow a desired trajectory. The other 2-DoF in the WMRA system are provided by the nonholonomic power wheelchair. The 2-DoF mobile platform consists of linear translation and rotation about a fixed axis. When controlling the mobile platform, velocities must be given for the linear translation as well as rotation. We use the weighted least-norm solution with singularity-robust pseudo inverse to resolve redundancies. As we will discuss later, we also use this weighted optimization to control coordination of the wheelchair platform and robotic arm during executed ADL tasks. Combination of the robotic arm and wheelchair kinematics is done using Jacobian augmentation, which can give the flexibility of using conventional control and optimization methods without compromising the total combined control [10]. Full kinematics and detailed equations can be found in a previous work concentrating on the control system [3].

The wheelchair will move forward when both wheels have the same speed and direction while rotational motion will be created when both wheels rotate at the same velocity but in opposite directions. Since the wheelchair's position and orientation are our control variables rather than the left and right wheels' velocities, a relationship between the wheels' rotational velocities and the linear and rotational velocities of the wheelchair was derived $(\dot{X}, \dot{\emptyset})$:

$$\begin{bmatrix} \dot{\theta}_l \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} \frac{1}{L_5} & \frac{-L_1}{2 \cdot L_5} \\ \frac{1}{L_5} & \frac{L_1}{2 \cdot L_5} \end{bmatrix} \cdot \begin{bmatrix} \dot{X} \\ \dot{\emptyset} \end{bmatrix} \quad (1)$$

Where L1 is the distance between the wheels and L5 is the wheels radius (see Figure 1). Seven DoFs are provided by the robotic arm mounted on the wheelchair from the Denavit-Hartenberg parameters of the robotic arm specified in earlier publications [3].

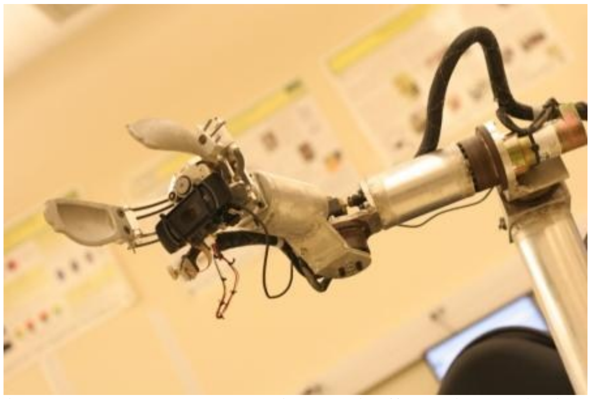

Fig. 2. USB camera mounted on end effector in eye in hand configuration.

B. Redundancy Resolution and Optimization

Redundancy is resolved in the algorithm using singularity-robust inverse of the Jacobian [20] to give a better approximation around singularities, and use the optimization for different subtasks. Manipulability measure [21] is used as a factor to measure how far the current configuration is from singularity.

Weighted Least Norm solution proposed by [11] is integrated to the control algorithm to optimize for secondary tasks. In order to put a motion preference of one joint rather than the other (such as the wheelchair wheels and the arm joints), a weighted norm of the joint velocity vector can be defined as:

$$|V|_W = \sqrt{V^T W V} \tag{2}$$

where *W* is a 9x9 symmetric and positive definite weighting matrix, and for simplicity, it can be a diagonal matrix that represent the motion preference of each joint of the system. The weighted least norm solution integrated to the S-R inverse is:

$$|V|_W = W^{-1}J^T(JW^{-1}J^T + k * I_6)^{-1}\dot{r} \tag{3}$$

where J is the augmented Jacobian of the WMRA system combining the robotic arm and wheelchair kinematics explained in [3], k is a parameter defined by the manipulability measure [3, 22] and ṙ represents the Cartesian velocities of the end effector.

The above method has been used in simulation of the 9-DoF WMRA system with the nine control variables (V) that represent the seven joint velocities of the arm and the linear and angular wheelchair's velocities. An optimization of criteria functions can be accomplished when used in the weight matrix *W*.

C. Criteria for Weighted Optimization

The criteria functions used in the weight matrix for optimization can be defined based on different requirements. For the robotic arm, the physical joint limits can be avoided by minimizing an objective function that represents this criterion. One of these mathematical representations was proposed by [11] as follows:

$$H(q) = \sum_{i=1}^{7} \frac{1}{4} \cdot \frac{(q_{i,max} - q_{i,min})^2}{(q_{i,max} - q_{i,current}) \cdot (q_{i,current} - q_{i,min})} \tag{4}$$

where q_i is the angle of joint i. This criterion function becomes 1 when the current joint angle is in the middle of its range, and it becomes infinity when the joint reaches either of its limits. The gradient projection of the criterion function can be defined as:

$$\frac{\partial H(q)}{\partial q_i} = \frac{(q_{i,max} - q_{i,min})^2 \cdot (2 \cdot q_{i,current} - q_{i,max} - q_{i,min})}{4 \cdot (q_{i,max} - q_{i,current})^2 \cdot (q_{i,current} - q_{i,min})^2} \tag{5}$$

When any particular joint is in the middle of the joint range, (5) becomes zero for that joint, and when it is at its limit, (5) becomes infinity, which means that the joint will carry an infinite weight that makes it impossible to move any further.

The diagonal weight matrix *W* can now be constructed as:

$$W = \begin{bmatrix} w_1 + \left|\frac{\partial H(q)}{\partial q_1}\right| & 0 & \dots & \dots & 0 \\ 0 & w_2 + \left|\frac{\partial H(q)}{\partial q_2}\right| & 0 & \dots & 0 \\ \vdots & 0 & \ddots & \dots & \vdots \\ \vdots & \vdots & \vdots & w_x & 0 \\ 0 & 0 & \dots & 0 & w_\varphi \end{bmatrix} \tag{6}$$

where w_i is a user-set preference value for each joint of the arm, and w_x and w_φ are the translation/rotation of the wheelchair.

D. Visual Servoing for Approach

At the beginning of the task, the user selects the goal object on a GUI, and then the wheelchair and arm work to center on the goal object and approach it. Coordination is controlled by criterion functions we define below. We initially set a rough velocity for the system depending on the distance to the goal object, and then control system movement with weights. We desire to initially use mostly wheelchair motion, but as we approach the goal object wheelchair motion should decrease as arm motion increases. We use an eye in hand monocular camera mounted on the end effector, as seen in Figure 2.

For the initial approach, it is only necessary for simple 2D visual servoing. We use camshift color tracking implemented in the OpenCV open source computer vision library for tracking the goal object [22]. Camshift returns the centroid of the matched template in the scene image, so in order to center on the selected area, we must adjust motion so that the tracked object's centroid reaches the center of the image plane, a=(c_u,c_v).

The wheelchair translation w_x is directly related to the distance from the camera frame to the goal object, in the camera frame's Z-direction. We approximate this distance by means of proximity sensor mounted on the end effector. Since w_x should be directly proportionate to the distance on Z, we have:

$$w_x = \frac{Z_i}{\lambda_1 Z} \tag{7}$$

where λ_1 is an appropriate gain, *Z* is the distance from the camera frame to the goal object estimated with a proximity sensor, and Z_i is the initial distance from the camera frame to the goal object.

The desired wheelchair rotation w_φ is directly related to the 2D visual servoing error. Since setting w_φ is only able to minimize the error in the camera frame's X-direction, we compute the error $e(t)_x$ using:

$$e(t)_x = s_x - c_u \tag{8}$$

where s_x is the current x-location of the centroid of the matched template, and c_u is the desired x-location of the template which is the center of the image plane in the camera frame's X-direction. Since w_φ should be directly proportionate to *e(t)*$_x$ computed in (8), we have:

$$w_\varphi = \frac{e(t)_{x\,max}}{\lambda_2 e(t)_x} \qquad (9)$$

where λ_2 is an appropriate gain and *e(t)*$_{x\ max}$ is the maximum possible error in the x-direction, in our case half of the image width in pixels.

We also desire to set the user-set preference values for w_1 through w_7 in order to control arm motion. We should use mostly wheelchair motion when the goal object is far away, and use mostly arm motion when the goal is very close. Therefore, we define the arm's user-set preference values for all 7 joints as:

$$w_1 = w_2 = \cdots = w_7 = \lambda_3 Z \qquad (10)$$

where λ_3 is an appropriate gain.

Using equations (7), (9), and (10) we can set motion so that the WMRA will approach the selected goal object area. Based on these criteria functions, initially when the distance Z is very large, primarily the wheelchair will move. As the WMRA approaches the goal object and z is reduced, the arm will begin moving as the wheelchair slows. Finally, when the WMRA has approached the goal object, strictly the arm will move. We then transition to grasping using IBVS when a defined threshold distance has been reached, while continuing use of the same weight equations (7), (9), and (10).

E. Visual Servoing for Grasping

We now implement an IBVS control system based on [13, 14] that outputs velocities to move the WMRA system until it has reached the 3D goal position and orientation. At this point, the gripper paddles can close and grasp the goal object, and the task is completed when the object is delivered to the user.

We desire to have a reliable and accurate method of feature extraction since the accuracy of the visual servoing control relies on this. Scale-invariant feature transform (SIFT), as described in [8], is a very robust feature extraction algorithm. For our code implementation, we use the open source SIFT library developed by Rob Hess in [23] to match features between the goal and scene images.

The goal of visual servoing is to minimize an error computed by:

$$e(t) = s(m(t), a) - s^* \qquad (11)$$

where *s(m(t),a)* are extracted features, *m(t)* is the vector of image measurements, and *a* is a set of camera parameters. In our case, *m(t)* consists of the image coordinates of the matched features in the scene image. From this point forward, we can represent *s(m(t),a)* simply as *s*. The vector s^* consists of the desired goal image measurements. In our case, s^* contains the image coordinates of the features in the goal image. Therefore, from (11), we see that the error *e(t)* is simply the difference between *s* and s^*.

For our application, we desire to design a velocity controller that can control the WMRA system using this visual servoing control. The relationship between the time variation of *s* and the camera velocity is described by:

$$\dot{\mathrm{s}} = \mathrm{L_s v_c} \qquad (12)$$

where $\mathrm{L_s}$ is the image Jacobian related to *s*, which we will define later. The vector $\mathrm{v_c}$ is the velocity controller for the WMRA system, which consists of v_c and ω_c, the instantaneous linear velocity and angular velocity, respectively, in all three dimensions. For visual servo control, $\mathrm{v_c}=(v_x, v_y, v_z, \omega_x, \omega_y, \omega_z)$. Using (11) and (12), we find the relationship between the time variation of the error and the camera velocity:

$$\dot{\mathrm{e}} = \mathrm{L_e v_c} \qquad (13)$$

where $\mathrm{L_e=L_s}$. We wish to solve (13) for $\mathrm{v_c}$ so that we can use it as velocity input to the WMRA control system. Therefore, we finally find:

$$\mathrm{v_c} = -\lambda_4 \widehat{\mathrm{L_e^+}} \mathrm{e} \qquad (14)$$

where λ_4 is a gain for the velocity control and the Moore-Penrose pseudo-inverse of $\mathrm{L_e}$ is taken to solve for $\mathrm{v_c}$.

We now define the image Jacobian to use in (14). We must first relate the 3D world point X=(X,Y,Z) to the 2D camera point x=(x,y):

$$\begin{aligned} x &= \frac{X}{Z} = (u - c_u) \\ y &= \frac{Y}{Z} = (v - c_v) \end{aligned} \qquad (15)$$

where m=(u,v) from (11) above is the coordinates in pixels of the image feature point, and a=(c_u,c_v) is the set of camera parameters with the principal point described by c_u and c_v. The image Jacobian is a 6x2k matrix for k matched feature points. The image Jacobian $\mathrm{L_x}$, related to x from (15) is:

$$\mathrm{L_x} = \begin{bmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{bmatrix} \qquad (16)$$

where x and y are from (15). In order to control the WMRA system using 6-DoF Cartesian control, we must have at least k=3 matched feature points to determine the velocities. We stack the image Jacobians for k points:

$$\mathrm{L_x} = [\mathrm{L_{x_1}} \quad \mathrm{L_{x_2}} \quad \cdots \quad \mathrm{L_{x_k}}]^{\mathrm{T}} \qquad (17)$$

Similarly, we also stack the errors such that e from (14) is:

$$\mathrm{e} = [\mathrm{e_{x_1}} \quad \mathrm{e_{y_1}} \quad \mathrm{e_{x_2}} \quad \mathrm{e_{y_2}} \quad \cdots \quad \mathrm{e_{x_k}} \quad \mathrm{e_{y_k}}]^{\mathrm{T}} \qquad (18)$$

We have now designed a visual servoing control system based on (14) from [13, 14] that can output velocity control for the WMRA. When the visual error has been minimized and the velocities of the system approach zero, then the robotic arm has reached its desired position and orientation. At this time, the gripper paddles can be closed to grasp the goal object and deliver it to the user in the wheelchair.

IV. Implementations and Results

Physical design of the 7-DoF manipulator and implementation onto the power wheelchair can be reviewed in [3, 4]. We use a Logitech C910 USB webcam mounted in eye in hand configuration on the end effector as seen in Figure 2. For estimating the z distance from the camera to the goal object, we use a Sharp GP2Y0A21YK infrared proximity sensor mounted just beneath the camera. Users operate the system with a laptop using a GUI developed for the application.

A. Go to and Pick Up Task

To demonstrate an example application of IBVS combined mobility and manipulation control, we execute a simple "Go to and Pick Up" task. Initially, the user is presented with a camera view, and once the object is selected it is tracked using camshift, and feedback is presented to the user in the GUI as seen in Figure 3. During the beginning of the approach, mostly the wheelchair moves. As the distance to the goal object decreases, the wheelchair slows and the arm begins moving. This can be seen visually with the weights displayed in Figure 4.

Once the end effector has reached a threshold distance from the goal object, SIFT-based IBVS begins and the user is presented with additional feedback in the GUI as seen in Figure 5. While grasping takes place, only manipulator movement is used to correctly position and orient it. Figure 6 shows the velocity output of IBVS while grasping. After IBVS finishes and the error is minimized, the gripper paddles then close to grasp the goal object, and it is delivered to the user through pre-programmed position control as seen in Figure 7.

Physical testing of this implementation generally results in a successful grasp. In 30 trials of the task, the system completed successfully 83.3% of the time. As can be seen from Figure 4, the system is able to control the weights to coordinate the combined mobility and manipulation during approach. Figure 6 shows that with the physical results of the system, the linear and angular velocities converge such that the error is minimized at the end of grasping. Although slight noise in the data exists, the physical system stays stabile during testing. In rare situations where the goal object was lost, the system halted all motion and prompted the user to re-select the goal object on the GUI.

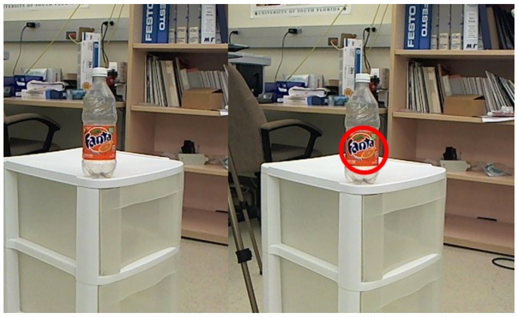

Fig. 3. GUI feedback while camshift object tracking during approach. The screen on the left is initially presented to the user to select the desired object, then it is tracked as seen on the right.

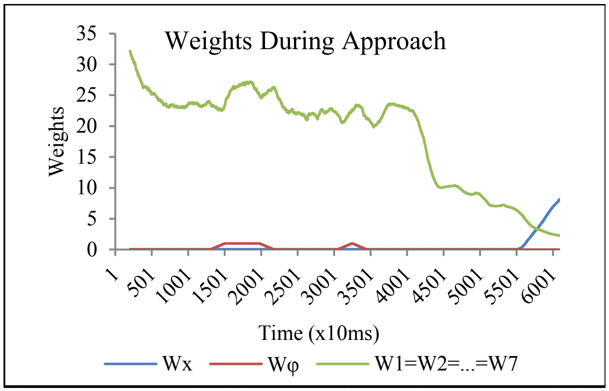

Fig. 4. Weights for wheelchair (w_x and w_φ) and arm ($w_1 \dots w_7$) during approach.

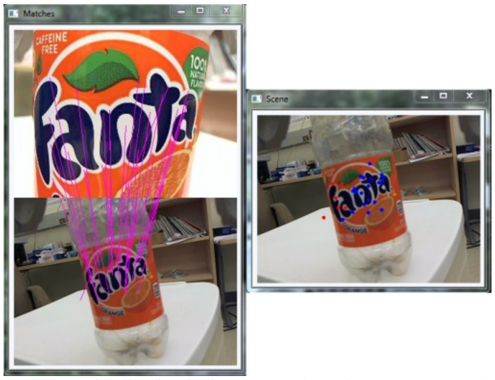

Fig. 5. GUI feedback during SIFT-based IBVS control while grasping. Rob Hess's open-source SIFT code [25] is used.

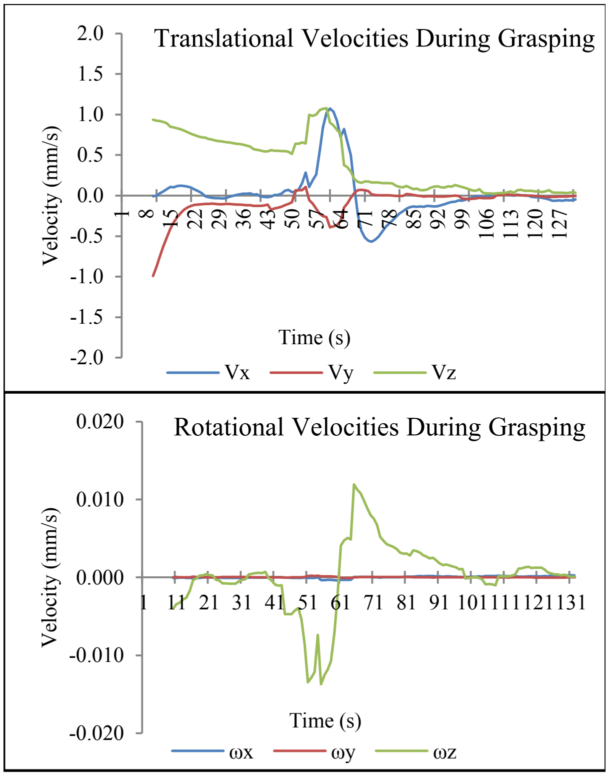

Fig. 6. Velocity output from visual servoing control while grasping. Translational velocity (top) is separated from rotational velocity (bottom) for easier viewing. Velocity reaches zero as the error is minimized (11), (14).

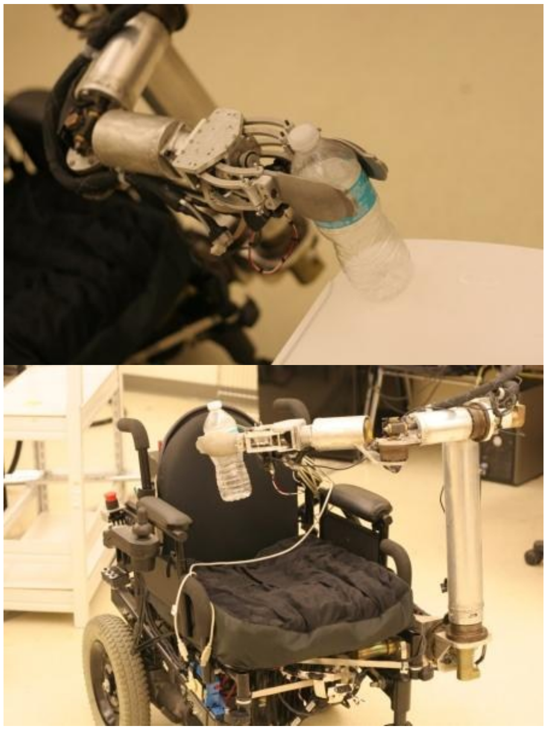

Fig. 7. Gripper grasping the goal object (top) at the end of IBVS when velocities reach zero, and then delivering the goal object to the user (bottom) through position control.

B. Go to and Open the Door Task

Previous work published in simulation presented the “Go to and Open the Door” task optimizing a second trajectory for the wheelchair while performing the main task with the end-effector [7]. Using a weight matrix, we successfully controlled the preference of motion for the arm or the rotation and translation of the wheelchair.

For the physical implementation, we have adopted the use of IBVS to compensate for the errors induced by the wheeled platform in the approach stage, giving a high robustness for the strategy aided by sensory information.

As with the task presented in the previous section, a weight for the arm was introduced to avoid over-stretching the arm while the WMRA is far away from the door knob.

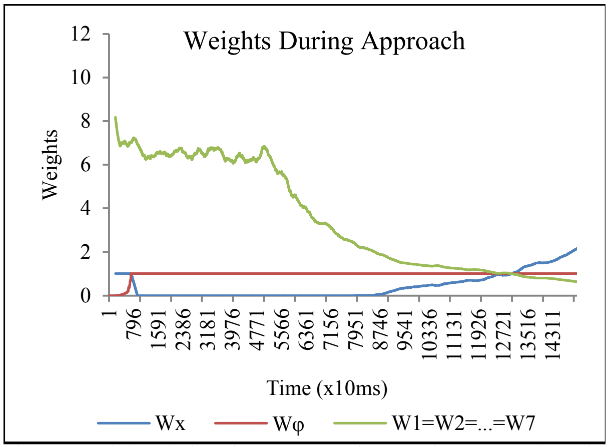

Fig. 8. Weights during the “Go to and Open the Door task.” The wheelchair first rotates to face the door, then it translates straight forward until it approaches the door.

As shown in Figure 8, the weights are controlled according to (7), (8), (9) and (10) allowing the WMRA to autonomously:

- Rotate the wheelchair towards the door
- Approach the door using IBVS to center on the door knob for grasping
- Adjust Orientation of the wheelchair and the end effector for opening the door
- Grasp the door knob and execute a circular trajectory to open the door

Note that in Figure 8 the rotational weight starts very close to zero and increases as the wheelchair orients toward the door, while translational weight behaves opposite to prioritize translation once the optimal orientation is reached [7].

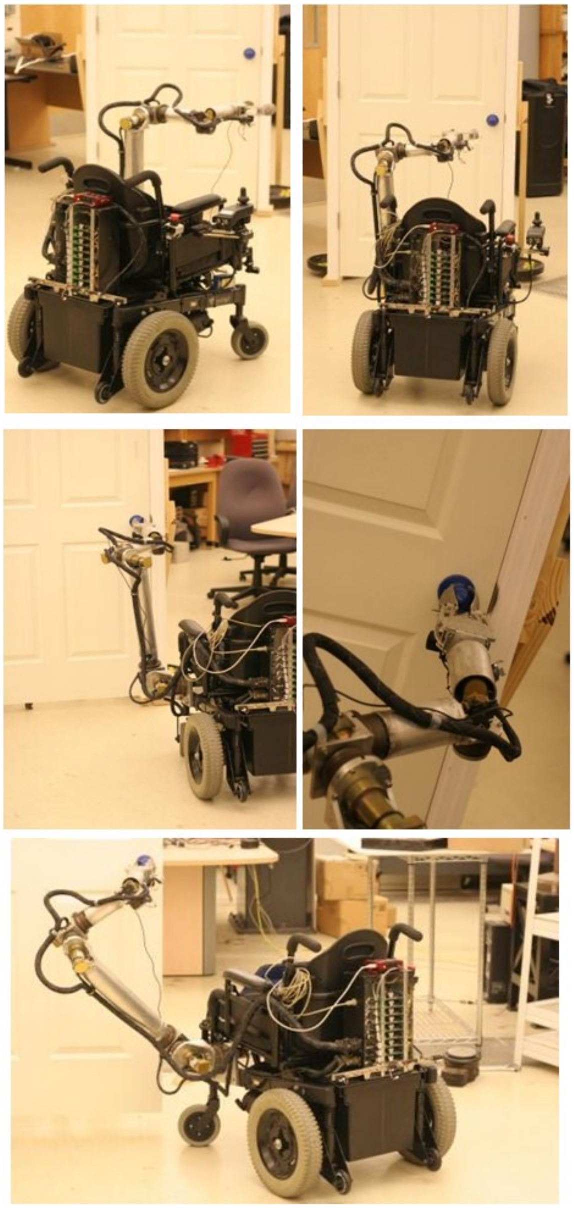

Fig. 9. WMRA approaching the door (up), adjusting and grasping the door knob (middle) and opening the door (down) to complete the “Go to and Open the Door” task.

V. Conclusion and Future Work

In this work, we have presented a novel approach and hardware implementation for combined mobility and manipulation of a WMRA system using visual servoing. We have also presented results from an assistive application for autonomous beginning to end execution of a "Go to and Pick Up" and "Go to and Open the Door" ADL task. Although this work is implemented on an assistive WMRA device, it could be extended to any mobile manipulator system.

Future work includes implementation of a potential fields collision avoidance system fused with the visual servoing velocity control. This would allow the system to navigate around obstacles autonomously. Other work includes the application of this work to other assistive ADL tasks.

References

[1] US Census Bureau, "Disability Among the Working Age Population: 2008 and 2009," Census Brief, September 2010, http://www.census.gov/prod/2010pubs/acsbr09-12.pdf

[2] Reswick J.B., "The Moon over Dubrovnik - A Tale of Worldwide Impact on Persons with Disabilities," *Advances in External Control of Human Extremities*, 1990.

[3] Alqasemi, R.; Dubey, R.; , "Maximizing Manipulation Capabilities for People with Disabilities Using a 9-DoF Wheelchair-Mounted Robotic Arm System," *Rehabilitation Robotics, 2007. ICORR 2007. IEEE 10th International Conference on* , vol., no., pp.212-221, 13-15 June 2007.

[4] P Schrock, P.; Farelo, F.; Alqasemi, R.; Dubey, R.; , "Design, simulation and testing of a new modular wheelchair mounted robotic arm to perform activities of daily living," *Rehabilitation Robotics, 2009. ICORR 2009. IEEE International Conference on* , vol., no., pp.518-523, 23-26 June 2009.

[5] Palankar, M.; De Laurentis, K.J.; Alqasemi, R.; Veras, E.; Dubey, R.; Arbel, Y.; Donchin, E.; , "Control of a 9-DoF Wheelchair-mounted robotic arm system using a P300 Brain Computer Interface: Initial experiments," *Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference on* , vol., no., pp.348-353, 22-25 Feb. 2009.

[6] C. Bringes, J. Jones, M. Malek, B. Simons, W. Pence, and R. Alqasemi. *Human-Computer Interaction for the WMRA*, FCRAR 2011.

[7] Farelo, F.; Alqasemi, R.; Dubey, R.; , "Optimized dual-trajectory tracking control of a 9-DoF WMRA system for ADL tasks," *Robotics and Automation (ICRA), 2010 IEEE International Conference on* , vol., no., pp.1786-1791, 3-7 May 2010.

[8] Lowe, David. "Distinctive Image Features from Scale-Invariant Keypoints". International Journal of Computer Vision 2004.

[9] Chengjun Ding; Ping Duan; Minglu Zhang; Huimin Liu; , "The kinematics analysis of a redundant mobile manipulator," *Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on* , vol., no., pp.2352-2357, 1-3 Sept. 2008.

[10] De Luca, A.; Oriolo, G.; Giordano, P.R.; , "Kinematic modeling and redundancy resolution for nonholonomic mobile manipulators," *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on* , vol., no., pp.1867-1873, 15-19 May 2006.

[11] Tan Fung Chan; Dubey, R.V.; , "A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators," *Robotics and Automation, IEEE Transactions on* , vol.11, no.2, pp.286-292, Apr 1995.

[12] R. Alqasemi. "Maximizing Manipulation Capabilities for People with Disabilities Using a 9-DoF Wheelchair-Mounted Robotic Arm System". Doctoral Dissertation. University of South Florida. Tampa, 2007.

[13] Chaumette, F.; Hutchinson, S.; , "Visual servo control. I. Basic approaches," *Robotics & Automation Magazine, IEEE* , vol.13, no.4, pp.82-90, Dec. 2006.

[14] Hutchinson, S.; Hager, G.D.; Corke, P.I.; , "A tutorial on visual servo control," *Robotics and Automation, IEEE Transactions on* , vol.12, no.5, pp.651-670, Oct 1996.

[15] Driessen, B.; Liefhebber, F.; Kate, T.T.; Van Woerden, K.; , "Collaborative control of the MANUS manipulator," *Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on* , vol., no., pp. 247- 251, 28 June-1 July 2005.

[16] Liefhebber, F.; Sijs, J.; , "Vision-based control of the Manus using SIFT," *Rehabilitation Robotics, 2007. ICORR 2007. IEEE 10th International Conference on* , vol., no., pp.854-861, 13-15 June 2007.

[17] Kim, Dae-Jin; Lovelett, Ryan; Behal, Aman; , "Eye-in-hand stereo visual servoing of an assistive robot arm in unstructured environments," *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on* , vol., no., pp.2326-2331, 12-17 May 2009.

[18] Ying Wang; Haoxiang Lang; de Silva, C.W.; , "A Hybrid Visual Servo Controller for Robust Grasping by Wheeled Mobile Robots," *Mechatronics, IEEE/ASME Transactions on* , vol.15, no.5, pp.757-769, Oct. 2010.

[19] Craig, J., 2003, "Introduction to robotics mechanics and control", third edition, Addison- Wesley Publishing, ISBN 0201543613.

[20] Nakamura, Y., "Advanced robotics: redundancy and optimization," Addison- Wesley Publishing, 1991, ISBN 0201151987.

[21] Yoshikawa, T., "Foundations of robotics: analysis and control," MIT Press, 1990, ISBN 0262240289.

[22] G.R. Bradski, Computer video face tracking for use in a perceptual user interface, Intel Technology Journal, Q2 1998.

[23] Rob Hess. An open-source SIFT Library. In *Proceedings of the international conference on Multimedia* (MM '10). ACM, New York, NY, USA, 1493-1496.