Tactics for using a Mobile Robot to Recover an Overturned wheeled Vehicle

Hidetoshi Ikeda (✉ ikedah@nc-toyama.ac.jp)

National Institute of Technology, Toyama College

Shinya Atoji

National Institute of Technology, Toyama College

Manami Amemiya

National Institute of Technology, Toyama College

Shingo Tajima

National Institute of Technology, Toyama College

Takayoshi Kitada

National Institute of Technology, Toyama College

Kotaro Fukai

National Institute of Technology, Toyama College

Keisuke Sato

National Institute of Technology, Toyama College

Research Article

Keywords: Tactics , Mobile Robot, Overturned wheeled Vehicle

Posted Date: November 18th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1055408/v1>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Tactics for using a mobile robot to recover an overturned wheeled vehicle

Hidetoshi Ikeda^{1,*}, Shinya Atoji¹, Keisuke Sato², Manami Amemiya^{1,+}, Shingo Tajima^{1,+}, Takayoshi Kitada^{1,+}, and Kotaro Fukai^{1,+}

¹National Institute of Technology, Toyama College, Department of Mechanical Engineering, Toyama, 939-8630, Japan

²National Institute of Technology, Toyama College, Department of Electrical and Control Systems Engineering, Toyama, 939-8630, Japan

*ikedah@nc-toyama.ac.jp

+these authors contributed equally to this work

ABSTRACT

This paper describes mobile robot tactics for recovering a wheeled vehicle that has overturned. If such a vehicle were to tip over backward off its wheels and become unable to recover itself, especially in areas where it would be difficult for humans to enter and work, total working efficiency could decline significantly, not only because the vehicle is no longer able to perform its job, but because it becomes an obstacle to other work. Herein, we propose a robot-based recovery method that can be used to recover such overturned vehicles and demonstrates its utility. The recovery robot, which uses a mounted manipulator and hand to recover the overturned vehicle, is also equipped with a camera and a personal computer (PC). The ARToolKit software package installed on the robot-mounted PC detects ARmarkers attached to the overturned vehicle and uses the information they provide to orient itself in order to perform recovery operations. To facilitate these ARToolKit-based operations, it is necessary to conduct a theoretical analysis to clarify the distance between the robot manipulator and vehicle, and thus achieve an objective position for grasping the overturned vehicle. The experimental results obtained in this study demonstrate the effectiveness of our proposed system.

Introduction

A number of developing countries are grappling with the serious twin problems of declining birthrates and aging populations, which are combining to reduce their total available workforce. In response to this, various parties have posited that if robots could be employed in dangerous areas and/or perform jobs that most people would not want to do because such work makes them feel mentally and physically distressed, the situation could be improved. Since many such jobs would require robots to move independently, various locomotive designs such as wheeled and tracked vehicles¹, special wheels², or the fusion of different mechanisms^{3,4}, have been researched widely. However, even though a great deal of effort has directed at improving the movement capabilities of mobile robots, none of the movement methods investigated thus far are perfect. Each method has its merits and demerits, and every movement mechanism has its limitations.

Wheel-based mechanisms are currently the most widely researched method for producing mobile robots. However, the primary problem with such mechanisms is their propensity to fall over or suffer from drive wheel slippage when it is necessary to navigate across rough terrain, run over obstacles, or when collisions occur with people or other robots. Overturning is especially problematic for wheeled robotic vehicles because such mishaps can result in fatal damage to the robot body. Hence, in cases where a robotic vehicle has overturned, the robot must be able to right itself, or it must be recovered some other way. If this is not done, not only will the vehicle be unable to perform its assigned duties, the vehicle itself becomes an obstacle that can prevent the free movement of humans or other robots. If such accidents occur areas where access limitations make it difficult for people or other robots to enter and work, the total workspace efficiency will decline significantly. Contrastingly, it is clear that work efficiency would be significantly improved in areas where multiple robots exist and work together at the same time if they were designed and constructed with the ability to assist each other when mishaps, such as overturned vehicles, occur.

Some research programs have already been conducted to investigate scenarios involving obstacle avoidance and the recovery of overturned humanoid robots^{5,6,7}. Other studies have examined ways to minimize damage when humanoid robots have overturned⁸ and a method of returning an overturned robot to a standing position has been studied⁹. For example, Benjamin Tam studied a fall avoidance and recovery method using a humanoid robot equipped with walking sticks¹⁰.

In the present paper, we describe the design and construction of a robot system that is capable of recovering a robotic vehicle that has overturned and placing it back on to its wheels. To accomplish this, the recovery robot must be able to raise the

Figure 1. Recovery robot

Figure 2. Model of the robot and overturned robotic vehicle

overturned robotic vehicle while continuously controlling its inclination. However, when the mass of the overturned robotic vehicle is heavier than that of the mass of the assisting robot, such a degree of control is difficult to achieve. That point was examined in Yoshida et al., and Inaba et al., who studied pivoting transportation methods using a humanoid robot¹¹¹². In a separate study, Asama et al. examined a transportation method using multiple robots that could raise an object¹³.

In our previous efforts, our research group investigated a step climbing method that is capable of inclining a robotic vehicle¹⁴ or a handcart¹⁵ in order to permit it to ascend a step.

The remainder of this paper is organized as follows. The next section describes our newly designed robot recovery system, the process of recovering a robotic vehicle that has fallen over backward, the results of our theoretical analysis and our experiments and their results. Section of Discussion contains our conclusions.

Result

Robot-based recovery system

The wheeled recovery robot used in this research, which will be referred to as the “robot” hereafter, was developed by our research group (Fig. 1). Table ?? in Appendix lists its specifications. Fig. 2 shows a model of the robot and overturned robotic vehicle. This robot has a manipulator that has two-degrees-of-freedom (2DOF) and an attached hand that has 1DOF, for a total of 3DOF. It is mounted on the center of the robot’s body. Encoders are set on each axis of the robot manipulator and on the hand mechanism. In this paper, the length of the upper arm link (from the shoulder joint to the elbow joint) is l_u , the length of the forearm link (from the elbow joint to the hand) is l_f . The angle of the shoulder joint is $-30 \text{ deg} \leq \phi_2 \leq +30 \text{ deg}$, and the angle of the elbow joint is $-90 \leq \phi_3 \leq +90 \text{ deg}$.

The robot hand is equipped with two fingers, and the fingers angle is $0 \text{ deg} \leq \xi \leq +50 \text{ deg}$. (See Fig. 3 (a).) The hand is equipped with an ultrasonic sensor (Digilent Pmod MAXSONAR) and an infrared sensor (POLORU VL6180X) for distance measuring, as shown in Fig. 3 (b). A switch sensor (OMRON, D2F-L) is installed inside the hand mechanism to detect contact information when the fingers are closed.

Figure 3. Robot hand mechanism: (a) fingers, (b) hand-mounted sensors.

Figure 4. Robot views: (a) Front view. (b) Rear view.

This robot has two pairs of wheels, and each pair consists of left and right wheels mounted in tandem. The front pair are casters, and the rear pair are individually-mounted driving wheels. The robot has motors and encoders on both driving wheel mechanisms that control its movements, as shown in Fig. 4 (a) and (b).

When the robot approaches the overturned robotic vehicle, the robot manipulator grasps a part of the vehicle using its hand mechanism (Fig. 2 and Fig. 3 (a)) and attempts to help restore the overturned vehicle to its correct working position. Table ?? in Appendix lists the specifications of the overturned robotic vehicle, which will be referred to as the "vehicle" hereafter in this paper (Fig. 5 (a) and (b)). Like the recovery robot, the vehicle also has a wheeled mechanism that consists of front and rear wheel pairs. The front pair are casters, and the rear pair are individually-mounted driving wheels.

Fig. 6 shows a configuration diagram for the system controls. The robot PC is equipped with an Intel (R) Core (TM) i3-7100U 2.40 GHz central processing unit (CPU) and 8GB of random access memory (RAM). The Linux (Lubuntu 18.04 LTS) operating system (OS) is installed. A Logicool HD Webcam C615 (Angle of view: 74 deg) is mounted on the front of the robot and connected to the PC. The ARtoolkit software package is installed on the PC.

ARtoolKit is a software package that is capable of perceiving posture data and calculating the distance between the camera and an ARmarker.¹⁶¹⁷¹⁸ ARtoolkit users are given free permission to make and use ARmarkers in their systems, providing that they obey the rules governing their manufacture. However, before using ARtoolkit, users must register the ARmarkers used in their systems.

In this study, ARmarkers are mounted on the front part (Fig. 5 (a), Size: 70 mm × 70 mm) and bottom (Fig. 5 (b), Size: 80 mm × 80 mm) of the overturned vehicle. The robot is able to perceive posture and distance information from the overturned vehicle based on data from these ARmarkers.

The robot is equipped with microcomputers (Arduino Holding, Arduino Leonardo) connected to the PC via Universal Serial Bus (USB) cables and connectors. Three motor driver circuits (Cytron Co., Ltd., MD10C) are connected to each microcomputer to control the driving processes. Electric motors (driving wheels: Tsukasa Electric Co., Ltd., TG-85E-SU-114-KA; shoulder and

Figure 5. Mounted ARmarkers on the robotic vehicle: (a) front, and (b) bottom.

Figure 6. System configuration

elbow axes: Tsukasa Electric Co., Ltd., TG-85E-KU-113-KA; and hand: HG16-240-AB-00) are connected to the motor driver circuits. Rotary encoders permit the degrees of these motors to be detected. (Driving wheels: Autonics, E30S4-100-3-N-5, shoulder axis, elbow axis, and hand mechanism: Taiwan Alpha Electronic, RE640F-40E3-20A-24P).

Overturned vehicle recovery process

Initially, to recover a vehicle that has fallen off its wheels, the robot approaches the overturned vehicle and grasps it using the hand mechanism attached to its manipulator arm. In this paper, it is assumed that the camera on the robot is able to capture the ARmarkers of the overturned vehicle and that both the robot and vehicle are on a flat surface.

The recovery process is described in detail below, and each stage of the process is shown in Fig. 7. The states shown in the figure correspond to (1) – (8) below:

(1) At this point, the vehicle has fallen over backward, and its incline is γ . The camera on the robot captures imagery of the overturned vehicle. ARtoolKit perceives the ARmarker mounted on the bottom of the vehicle's body and calculates the relative distance between the robot and the vehicle, as shown in Fig. 5 (b) .

(2) The robot drives its left and right wheels to pivot around its center position until it is oriented towards the overturned vehicle. Once the relative angle between the robot-mounted camera and the overturned vehicle is reduced to approximately zero, ARtoolKit calculates the distance between the robot and the overturned vehicle.

(3) The robot moves toward the overturned vehicle and stops after traveling half of the distance between it and the vehicle.

(4) Next, the robot again pivots on its left and right rear wheels until the relative angle between its front-mounted camera and the vehicle is reduced to approximately zero, after which ARtoolKit again determines the distance between the robot and

Figure 7. Overturned vehicle recovery process.

the vehicle.

These processes (ARmarker perception, pivot turning, and forward motion) are repeated until the distance between the robot and the overturned vehicle is reduced to 0.3 m.

(5) Once the distance between the robot hand and the overturned vehicle declines to 0.3 m, the ultrasonic sensor on the hand searches for and detects the front bar of the vehicle. Next, the infrared distance sensor on the hand detects the front bar of the vehicle and performs minor position adjustments until the robot hand is able to grasp the front bar of the overturned vehicle (Fig. 3) and then stops. At that point, the robot hand grasps the objective position on the front bar of the vehicle.

(6) Next, the robot moves backward, causing the incline of the overturned vehicle, γ , to increase. The connection position between the robot and the vehicle begins moving up along the front bar of the overturned vehicle. At that point, the robot uses ARtoolKit to monitor both the incline of the vehicle and the ARmarker mounted on its bottom, as shown in Fig. 5 (b).

(7) The robot continues to move backward. When the γ is more than 120 deg, the robot perceives the other ARmarker mounted on the front part of the vehicle, as shown in Fig. 5 (a).

(8) The robot continues to move backward until it perceives $\gamma = 180$ deg, and then stops. At that point, the vehicle has been recovered from the overturned state.

Theoretical analysis of the distance between the robot hand and the overturned vehicle

Fig. 8 (a) shows the initial situations of the robot and overturned vehicle. The basic robot coordinate system is denoted by Σ_{B0} , where point B is the origin. Here, Point B is the center between the two floor surface contact positions for the left and right rear wheels. In this figure, x indicates the roll axis for the initial direction of the robot, y indicates the yaw axis, and z indicates the pitch axis. C_0 indicates the camera mounting position on the robot in Σ_{B0} .

β is the angle, which is formed by the x axis and the line from Point B to Point M (ARmarker position) on the xz plane. Similarly, β_C is the relative angle, which is formed by the x axis and the line from C_0 to M on the xz plane.

When the robot turns toward the overturned vehicle, the Σ_{B0} , x axis, and z axis change to Σ_B , x' axis, and z' axis, respectively (See Fig. 8 (b)). The camera position on the robot, C_0 , changes to C .

Fig. 9 shows the situation that results after a pivot turn when the robot is oriented toward the overturned vehicle. The local coordinate system of the camera mounted on the robot is denoted as Σ_C , which always parallels Σ_B . The position vector for the camera in Σ_B is expressed as ${}^B p_C$.

Figure 8. Model showing the robot and overturned vehicle: (a) the relative angle, and (b) the robot oriented towards the vehicle.

Figure 9. Model showing the overturned vehicle recovery.

The camera position, ${}^B\mathbf{p}_C$, is described below (Fig. 2),

$${}^B\mathbf{p}_C : \begin{bmatrix} x_C \\ y_C \\ z_C \end{bmatrix} = \begin{bmatrix} l_{CB} \\ R_B + h_{CB} \\ 0 \end{bmatrix} \quad (1)$$

ϕ_i is the angle of local coordinate systems (Σ_i), Σ_i is formed by Σ_{i-1} . In this system, both the inclination of the robot and ϕ_1 is zero. The position vectors for these joints in system Σ_B are expressed as ${}^B\mathbf{p}_i = [x_i \ y_i \ z_i]^T$ ($i = 1 - 4$), where the robot hand position, ${}^B\mathbf{p}_4$, which is described in Fig. 2 below.

$${}^B\mathbf{p}_4 : \begin{bmatrix} x_4 \\ y_4 \\ z_4 \end{bmatrix} = \begin{bmatrix} l_{fL} \cos \phi_{23} + l_{uL} \cos \phi_2 + l_{LB} \\ l_{fL} \sin \phi_{23} + l_{uL} \sin \phi_2 + h_{LB} + R_B \\ 0 \end{bmatrix} \quad (2)$$

Here, $\phi_{23} = \phi_2 + \phi_3$.

As shown in Fig. 9, the incline of the overturned vehicle is γ , and its local coordinate system is denoted as Σ_A . Here, Point A is the center between the two surface contact positions for the left and right rear wheels, Point M is the position of the ARmarker mounted on the vehicle bottom, and Point F is the objective position for the robot grasping the vehicle. ${}^A\mathbf{p}_M = [{}^Ax_M \ {}^Ay_M \ {}^Az_M]^T$ is the ARmarker position vector in Σ_A , which is described below.

$${}^A\mathbf{p}_M : \begin{bmatrix} {}^Ax_M \\ {}^Ay_M \\ {}^Az_M \end{bmatrix} = \begin{bmatrix} l_M \cos \gamma + h_M \sin \gamma \\ l_M \sin \gamma - h_M \cos \gamma + R_A \\ 0 \end{bmatrix} \quad (3)$$

${}^A\mathbf{p}_F$ is the position vector of point F in Σ_A . It is expressed as,

$${}^A\mathbf{p}_F : \begin{bmatrix} {}^Ax_F \\ {}^Ay_F \\ {}^Az_F \end{bmatrix} = \begin{bmatrix} l_F \cos \gamma + h_F \sin \gamma \\ l_F \sin \gamma - h_F \cos \gamma + R_A \\ 0 \end{bmatrix} \quad (4)$$

The position vector of the ARmarker in Σ_C is ${}^C\mathbf{p}_M$

$${}^C\mathbf{p}_M = \begin{bmatrix} {}^Cx_M \\ {}^Cy_M \\ {}^Cz_M \end{bmatrix} \quad (5)$$

${}^C\mathbf{p}_M = [{}^Cx_M \ {}^Cy_M \ {}^Cz_M]^T$ is gotten using ARToolKit.

When the robot is oriented towards the overturned vehicle, ${}^Cz_M = 0$. In this case,

$${}^C\mathbf{p}_M = \begin{bmatrix} {}^Cx_M \\ {}^Cy_M \\ 0 \end{bmatrix} \quad (6)$$

Geometrically, we get Equation (7), as shown in Fig. 8 (a),

$$\tan \beta_C = \frac{{}^Cz_M}{{}^Cx_M} \quad (7)$$

Using ARtoolKit, the robot conducts a pivot turn until the value of β_C (Fig. 8 (a)) becomes approximately zero, and it is oriented towards the overturned vehicle. In this case, the position vector of A in Σ_B is expressed as ${}^B\mathbf{p}_A$, as shown in Fig. 9,

$${}^B\mathbf{p}_A = {}^B\mathbf{p}_C + {}^C\mathbf{p}_M - {}^A\mathbf{p}_M \quad (8)$$

From (1), (3), (6),

$${}^B\mathbf{p}_A = \begin{bmatrix} l_{CB} + {}^Cx_M - (l_M \cos \gamma + h_M \sin \gamma) \\ R_B + h_{CB} + {}^Cy_M - (l_M \sin \gamma - h_M \cos \gamma) \\ 0 \end{bmatrix} \quad (9)$$

The vector from the robot hand to the objective position where it will grasp the overturned vehicle is ${}^4\mathbf{p}_F = [{}^4x_F \ {}^4y_F \ {}^4z_F]^T$,

$${}^4\mathbf{p}_F = -{}^B\mathbf{p}_4 + {}^B\mathbf{p}_A + {}^A\mathbf{p}_F \quad (10)$$

From (2), (4), (9),

$${}^4\mathbf{p}_F : \begin{bmatrix} {}^4x_F \\ {}^4y_F \\ {}^4z_F \end{bmatrix} = \begin{bmatrix} c_{x_M} - H_x + l_C + (l_F - l_M) \cos \gamma - (h_M - h_F) \sin \gamma \\ c_{y_M} - H_y + h_C + (l_F - l_M) \sin \gamma + (h_M - h_F) \cos \gamma + R_A \\ 0 \end{bmatrix} \quad (11)$$

Here, $H_x = l_{fL} \cos \phi_{23} + l_{uL} \cos \phi_2 + l_{LB}$, $H_y = l_{fL} \sin \phi_{23} + l_{uL} \sin \phi_2 + h_{LB}$.

When the robot identifies the objective grasping position of the overturned vehicle, it can set the hand at the same height as point F on the vehicle. In this case, ${}^4y_F = 0$ in Equation (11), and the distance between the hand and point F , $|{}^4\mathbf{p}_F|$, are expressed as,

$$|{}^4\mathbf{p}_F| = |{}^4x_F| = \sqrt{\{c_{x_M} - H_x + l_{CB} + (l_F - l_M) \cos \gamma - (h_M - h_F) \sin \gamma\}^2} \quad (12)$$

The robot moves in order to bring $|{}^4\mathbf{p}_F|$ to zero using its sensors and ARtoolKit. At that point, the robot can grasp the overturned vehicle.

Experiment

An experiment was carried out in an environment with a surface friction coefficient of $\mu = 0.68$. The robot and overturned vehicle were placed on the same floor within the National Institute of Technology, Toyama College. The maximum movement speed of the robot was set at 3.6 km/h.

Figure 10. Position arrangement and angle of overturned vehicle: (a) position, and (b) angle.

We began by conducting experiments to ensure the ARmarker could be detected using the robot-mounted camera. From the results obtained, we found that ARtoolkit sometimes reacted incorrectly in situations where the camera captured lattice-like designed furniture or posters. In those situations, the robot could not be properly controlled. Accordingly, the external environment in which the robot and vehicle experiments were conducted had a white wall and was further partitioned from the rest of the room by suspending a piece of white cloth between the robot and any potentially distracting objects.

We then determined that the camera could perceive the ARmarker mounted on the overturned vehicle at distances up to 1.5 m and that when the robot and the vehicle were separated by less than 1.5 m, the camera was able to capture the ARmarker's position in $-0.4 \text{ m} \leq z_A \leq 0.4 \text{ m}$, as shown in Fig. 10 (a), (b). Furthermore, when the relative angle between the ARmarker and the camera, θ , was within $-35 \text{ deg} \leq \theta \leq 35 \text{ deg}$, the camera could perceive the ARmarker. The body angle of the overturned vehicle, γ , could be detected within $78 \text{ deg} \leq \gamma \leq 136 \text{ deg}$.

Thus, the recovery experiments were carried out from the overturning point shown in Fig. 10. Here, $x_A = 1.5 \text{ m}$, $z_A = 0.4 \text{ m}$, the relative angle between ARmarker and the camera is $\theta = 90 \text{ deg}$, and the first incline of the vehicle is set at $\gamma = 90 \text{ deg}$. The robot perceived the ARmarker mounted on the vehicle bottom within $\gamma = 90 \text{ deg} \leq \gamma \leq 120 \text{ deg}$, as shown in Fig. 5

Figure 11. Experiment

(b), and when the incline of the vehicle was over 120 deg, the robot perceived the ARmarker mounted on the vehicle front, as shown in Fig. 5 (a).

We assumed that the robot had already determined the correct position to grasp the vehicle to effect a successful recovery ($h_F = 20$ mm, $l_F = 320$ mm). We also assumed that the driving wheels of the robot and vehicle would not slip, and that the robot could generate sufficient force to pull and support the overturned vehicle during the recovery process.

The states shown in Fig. 11 correspond to (1) – (8). Here, each figure on the left side shows movements of the robot and vehicle, while each figure on the right side shows video imagery captured by the camera mounted on the robot. The arrows shown in the right side figures of Fig. 11 indicates that the ARtoolkit was able to perceive the ARmarkers.

The camera on the robot captured video imagery of the vehicle at the screen center. Then, based on its perception of the ARmarker mounted on the vehicle bottom, the robot repeated its pivot turn to correct its orientation with the overturned vehicle. The robot then traveled half of the distance between it and the vehicle, stopped, and repeated the ARmarker perception, pivot turn, and forward motion processes until the robot hand was able to grasp the front bar of the vehicle.

Once the front bar was grasped, the robot began moving backward, exerting a pulling force on the overturned vehicle, and causing γ to incline. Up to that point, the robot's actions were based on its perception of the ARmarker mounted on the vehicle bottom. However, once γ exceeded 120 deg, the robot perceived the vehicle's front-mounted ARmarker instead of the bottom

ARmarker. The robot continued moving backward until the $\gamma = 180$ deg, at which time it stopped. At that point, the vehicle was back on its wheels, and the recovery process was complete.

As described above, it was necessary to modify the external environment in which the robot and vehicle operated during the experiment to make it suitable for ARtoolKit use. This required removing or hiding objects with lattice-like designs. Once this was accomplished, the robot was able to work stably and successfully recovery the overturned vehicle.

Discussion

This report described potential recovery tactics for use when employing a mobile robot to recover a vehicle that had fallen over backward and showed how such a robot system could be constructed. A theoretical analysis clarified how resolving the distance between the robot and the overturned vehicle made it possible for the robot to grasp the vehicle and return it to an upright position.

Since our results showed that ARtoolkit sometimes reacted incorrectly when the robot's camera captured furniture or posters with lattice-like designs, such items were removed or covered, and the experiments in this study were conducted in an environment that allowed ARtoolKit to correctly perceive the overturned vehicle. In those circumstances, the robot was able to work stably and recovery the overturned vehicle. Despite this restriction, the experimental results obtained confirmed the effectiveness of our proposed method. Therefore, it is thought that by further developing this system, it would be possible to establish a new technology that would allow vehicles to be easily recovered when an accident occurs. This would be particularly useful if such vehicles became incapacitated in areas that are inaccessible to humans.

In the future, we will conduct experiments under various other environments and work to improve the robot's perception capabilities. We will also construct a judgment system for determining the most suitable grasping positions when recovering an overturned vehicle, and will study other recovery methods to be used when a vehicle overturns or becomes stuck due to driving wheel slippage when traversing rough terrain.

References

1. Okada,Y., Nagatani,K. & Yoshida,K. Semi-autonomous operation of tracked vehicles on rough terrain using autonomous control of active flippers. In: Proceedings of 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2815-2820, doi: 10.1109/IROS.2009.5354549 (2009)
2. Taguchi,K. Enhanced wheel system for step climbing, ADVANCED ROBOTICS, **9**(2), 137-147, <https://doi.org/10.1163/156855395X00049> (1995).
3. Nakajima,S., Nakano,E. & Takahashi,T. Free gait algorithm with two returning legs of a leg-wheel robot. Journal of Robotics and Mechatronics, **20**(4), 661-668, <https://doi.org/10.20965/jrm.2008.p0661> (2008).
4. Kumar,V. & Krovi,V., Optimal Traction Control In A Wheelchair With legs And Wheels. In: Proceedings of 4th National Applied Mechanisms and Robotics Conference, 95-030-01 - 95-030-07 (1995)
5. Yun,S., Goswami,A., & Sakagami, Y., Safe fall: Humanoid robot fall direction change through intelligent stepping and inertia shaping. In: Proceedings of 2009 IEEE International Conference on Robotics and Automation, 781-787, <http://dx.doi.org/10.1109/ROBOT.2009.5152755> (2009)
6. Goswami, A., Yun,S., & Nagarajan, U., Direction-changing fall control of humanoid robots: theory and experiments, Autonomous Robot, **36**, 199–223, <https://doi.org/10.1007/s10514-013-9343-2> (2014)
7. Nagarajan,U. & Goswami,A., Generalized direction changing fall control of humanoid robots among multiple objects, In: Proceedings of 2010 IEEE International Conference on Robotics and Automation, 3316–3322 (2010).
8. Fujiwara,K. et al., UKEMI: falling motion control to minimize damage to biped humanoid robot, In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2521–2526, doi: 10.1109/IRDS.2002.1041648 (2002)
9. Fujiwara,K. et al., Falling motion control of a humanoid robot trained by virtual supplementary tests, In: Proceedings of IEEE International Conference on Robotics and Automation, doi: 10.1109/ROBOT.2004.1307968, 1077–1082 (2004)
10. Tam,B., & Kottege,N, Fall avoidance and recovery for bipedal robots using walking sticks, In: Proceedings of Australasian Conference on Robotics and Automation 2016, 225–233 (2016)
11. Murooka,M. et al., Achievement of Pivoting Large and Heavy Objects by Life-sized Humanoid Robot based on Online Estimation Control Method of Object State and Manipulation Force (In Japanese), Journal of Robotics Society of Japan, **32**(7), 595–602, <https://doi.org/10.7210/jrsj.32.595> (2014)
12. Yoshida,E. et al., Pivoting based manipulation by a humanoid robot, Autonomous Robot, **28**(77), <https://doi.org/10.1007/s10514-009-9143-x> (2010)

13. Yamashita,A., Arai,T, Ota,J. & Asama,H., Motion planning of multiple mobile robots for Cooperative manipulation and transportation. *IEEE Transactions on Robotics and Automation*, **19**(2), 223–237, doi: 10.1109/TRA.2003.809592 (2003)
14. Ikeda,H., Toyama,T., Maki.D., Sato,K. & Nakano,E. Cooperative step-climbing strategy using an autonomous wheelchair and a care Robot, *Robotics and Autonomous Systems*, **135**, 103670, 1–11, <https://doi.org/10.1016/j.robot.2020.103670> (2021)
15. Ikeda,H., Kawabe,T., Wada,R. & Sato,K. Step-Climbing Tactics Using a Mobile Robot Pushing a Hand Cart, *Applied Sciences*, **8**(11), 2114, <https://doi.org/10.3390/app8112114> (2018)
16. Kato,H., Ishida, M. & Billingham,M., Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System, In: *Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality '99'* 85–94, doi: 10.1109/IWAR.1999.803809 (1999)
17. Kato, H., Tachibana,K., Tanabe,M., Nakajima,T. & Fukuda,Y., MagicCup: a tangible interface for virtual objects manipulation in table-top augmented reality, In: *Proceedings of 2003 IEEE International Augmented Reality Toolkit Workshop*, Tokyo, Japan, 75–76, doi: 10.1109/ART.2003.1320434 (2003).
18. Piekarski,W. & Thomas,B.H., Using ARToolKit for3D hand position tracking in mobile outdoor environments. In: *Proceedings of the First IEEE International Workshop Augmented Reality Toolkit*, doi: 10.1109/ART.2002.1107003 (2002)

Acknowledgment

This research has been supported by the Artificial Intelligence Research Promotion Foundation (28AI 043-1), Japan.

Author contributions statement

Conceptualization: H.I; Methodology: H.I; Formal analysis and investigation: H.I; Writing - original draft preparation: H.I; Writing - review and editing: H.I; Funding acquisition: H.I; Software: S.A, M.A, K.S; Experiment: S.A, S.T, T.K, K.F; Supervision: H.I, K.S; Project administration: H.I

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Appendix.pdf](#)
- [captionexperimentmovie.mp4](#)