# Sway Motion Cancellation Scheme Using a RGB-D Camera-based Vision System for Humanoid Robots

Jeong-Ki Yoo, Seung-Beom Han, and Jong-Hwan Kim

**Abstract** When a humanoid robot walks dynamically, it generates sway motion which is reflected as an oscillative sine wave-like pattern at its center-of-mass (CoM) trajectory. In order to cancel out such motion from the coordinates of detected obstacles, this paper proposes a sway motion cancellation scheme incorporated with walking pattern generator of humanoid robots along with a RGB-D camera-based vision system. After the preprocessing for the depth information from the RGB-D camera using attitude reference system (ARS)-generated roll and pitch angles of the vision module, the coordinates of detected obstacles are estimated using the ground filtered 3D points. Then, the sway motion cancellation scheme is applied to the coordinates of detected obstacles not only for the lateral direction of the robot but also for the sagittal one by referring the CoM trajectory collected from the walking pattern generator. The proposed sway motion cancellation scheme and the RGB-D camera-based vision system are verified by experiments using a small-sized humanoid robot, HanSaRam-IX (HSR-IX).

**Key words:** Humanoid robot navigation, sway motion cancellation, depth camera-based vision module, RGB-D sensor.

## 1 Introduction

The researches for humanoid robots have been performed mainly focusing on the walking issues [1–4]. By the virtue of such researches to generate dexterous walking

Jeong-Ki Yoo
Digital Media and Communications R&D Center, Samsung Electronics, Suwon 443-742, Republic of Korea, e-mail: yoojeongki@gmail.com

Seung-Beom Han and Jong-Hwan Kim
Department of Electrical Engineering, KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon 305-701, Republic of Korea, e-mail: {sbhan, johkim}@rit.kaist.ac.kr

patterns, various fields of researches for mobile robots such as vision processing, navigation, task planning and human-robot interaction, etc. has been employed for humanoid robots [5–8]. In addition, due to the limited capacity of visual perception of humanoid robot, gaze control researches were also performed [9, 10]. There was an approach employing fuzzy measure and fuzzy integral for the gaze control of humanoid robots [11].

Due to the trait of walking process, humanoid robots keep swinging mainly along the lateral direction. For the precise acquisition of information for surrounding environment, such sway motion-originated side effect in the coordinates of objects have to be canceled out in advance before starting the vision processing. If the vision processing is performed without sway motion cancellation, oscillative CoM patterns would be reflected in each estimated coordinate of a detected obstacle whenever the robot walks. Thus, the sway motion cancellation is the one of essential preprocessing procedures for robust navigation of humanoid robots. Dune et al. proposed a sway motion cancellation strategy for visual servoing [12]. They used the predictive walking pattern generator information as reference signals to cancel out sway motion in Y-axis of HRP-2 robot. Note that they only focused on lateral direction of walking without considering sagittal one. Started from this research, the vision-based 3D model tracking research considering sway motion cancellation was also performed [13]. In the research, the previously proposed sway motion cancellation framework was adopted to match 3D object models with vision information. However, as they did not considered sway motion cancellation in the sagittal direction. Moreover, their approach required a precise prediction model due to the synchronization inaccuracy between components of their navigation system.

This paper proposes a sway motion cancellation scheme incorporated with humanoid walking pattern generation algorithm along with RGB-D camera-based vision system. As a RGB-D camera, a Microsoft Kinect sensor is used in this paper. HSR-IX, which was developed at the Robot Intelligence Technology laboratory, KAIST, uses modifiable walking pattern generation (MWPG) method to generate walking patterns [4, 14]. By using the center-of-mass (CoM) trajectory obtained from the walking pattern generator in the robot, the side effect caused by the sway motion during walking can be canceled out from the coordinates of detected obstacles which are obtained by the RGB-D camera-based vision system. In addition, the proposed sway motion cancellation scheme considers sagittal direction of walking as well as lateral one by The proposed scheme and system are verified by experiments using HSR-IX.

This paper is organized as follows. Section 2 explains how the vision system is implemented using a RGB-D camera. Section 3 presents the proposed sway motion cancellation scheme for humanoid robots. Then, Section 4 describes the result of experiments along with the developed experimental system, and concluding remarks follow in Section 5.

## 2 RGB-D Camera-based Vision System

In order to implement a vision system which is capable of providing depth information as well as color information, a RGB-D camera is employed as a vision sensor incorporated with attached ARS sensor to compensate roll and pitch angles of the vision module as shown in Fig. 1 [15–17]. The roll and pitch angles of the vision system against the gravity direction, $\boldsymbol{\theta}_{ars} = [\theta_r, \theta_p]^T$, are used to get rid of the 3D points belong to the ground by compensating the rotation of the vision module [17].

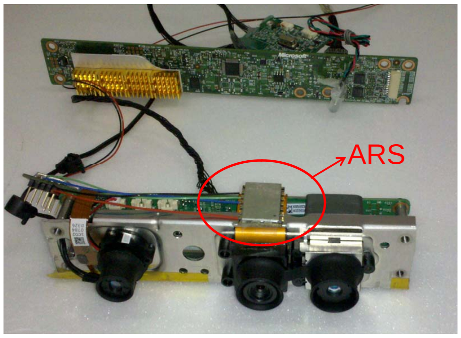


Fig. 1: Vision module with attached ARS sensor.

$\boldsymbol{\theta}_{ars}$ is obtained through a Bluetooth connection between the ARS sensor and a Linux-based vision processing server which calculates the coordinates of obstacles using OpenNI and OpenCV libraries [18, 19]. Since the conversion process of a RGB-D point to one in the world coordinate one using OpenNI library have to be applied point by point, it takes about 23 ms for converting a 640 x 480 resolution RGB-D image using an Intel CoreDuo 2.3 GHz computer with 2G RAM. To decrease this world coordinate conversion time, a look-up table for depth conversion is pre-generated and used. By using this look-up table approach, depth conversion time for one image could be decreased under 9 ms in average. Fig. 2 describes the vision processing procedure including ground filtering, color-based obstacle detection, and the generation of coordinates for detected obstacles from robot orientation.

After the preprocessing to eliminate the effect of radial distortion using the intrinsic matrix of RGB camera, color-based segmentation with the obstacle color set is performed using equally-spaced seeds for adaptive flooding of segments [19]. Then, the derived center points $\mathbf{c}_i$ of thresholded segments are converted to ones in the real world coordinate, $\mathbf{c}_i^R$ using the rotation matrix $R(\boldsymbol{\theta}_{ars})$ and the world coordinate generation function of OpenNI [18]. Incorporated with this color image-based process, depth image-based vision processing is also performed simultaneously. Note that the initial calibration to match the centers of color and depth-based images is performed in advance as shown in Fig. 2. Then the 3D points in the depth image,

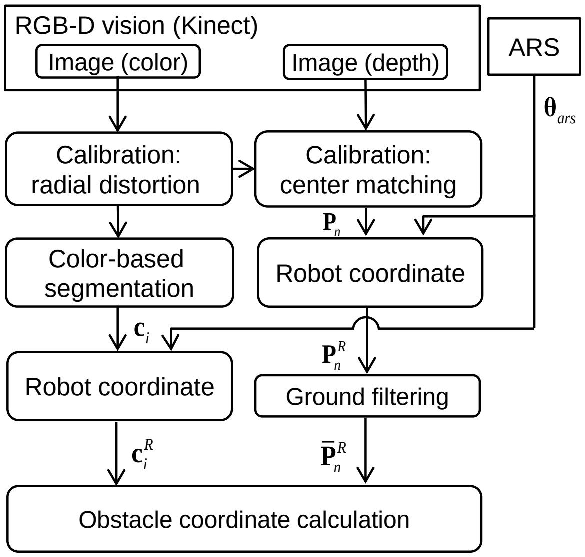


Fig. 2: Vision processing procedure.

$\mathbf{P}_n$, are converted to ones in the robot coordinate system, $\mathbf{P}_n^R$ by using the intrinsic parameters obtained from the process of the calibration [20]. After that, the ground points are filtered out to decrease computation time. That is, ones under the threshold are filtered out to obtain obstacle-related 3D points. In case of the HSR-IX, the threshold is set to -52 cm since the height of the robot is 52 cm. After that, the filtered points are moved by 52 cm along the Z-axis in order to compensate for the height of the robot. As a result, 3D points possibly belonging to obstacles, $\bar{\mathbf{P}}_n^R$, could be distinguished from the ground points. Since cylinder-shaped obstacles are used for this system, their center positions could be obtained by using the trait of the shape. The coordinates of obstacles are calculated as follows:

$$\hat{\mathbf{P}}_{o_i}^R = T(mean(\left\|\mathbf{Pset}_{o_i}^{near}\right\|_2), r_{obs}), \tag{1}$$

with

$$\begin{aligned}
&\mathbf{Pset}_{o_i}^{near} = \{\bar{\mathbf{P}}_n^R | \left\|\bar{\mathbf{P}}_{c_i}^R - \mathbf{P}_{n_{\min}}^R\right\|_2 < \lambda, \bar{\mathbf{P}}_{n_{\min}}^R \in \mathbf{A}_i\},\\
&n_{min} = \arg\min_n(\left\|\bar{\mathbf{P}}_n^R\right\|_2, \bar{\mathbf{P}}_n^R \in \mathbf{A}_i),\\
&\mathbf{A}_i = \{\bar{\mathbf{P}}_n^R | \left\|\mathbf{c}_i^R - \bar{\mathbf{P}}_n^R\right\|_2 < \alpha r_o, col(\bar{\mathbf{P}}_n^R) = i\},
\end{aligned}$$

where $\alpha$ is an initial threshold coefficient used for adjusting the boundary to select 3D points of the corresponding obstacle, $n_{min}$ is the index of the nearest point from the robot among $\mathbf{A}_i$, $\lambda$ means the small distance threshold for selecting closer points than $\alpha r_o$, mean($\beta$) calculates the mean coordinate of $\beta$, $r_o$ is the radius of obstacles, $col(C)$ checks the color index of corresponding 3D point $C$, and $T(D,E)$ translates

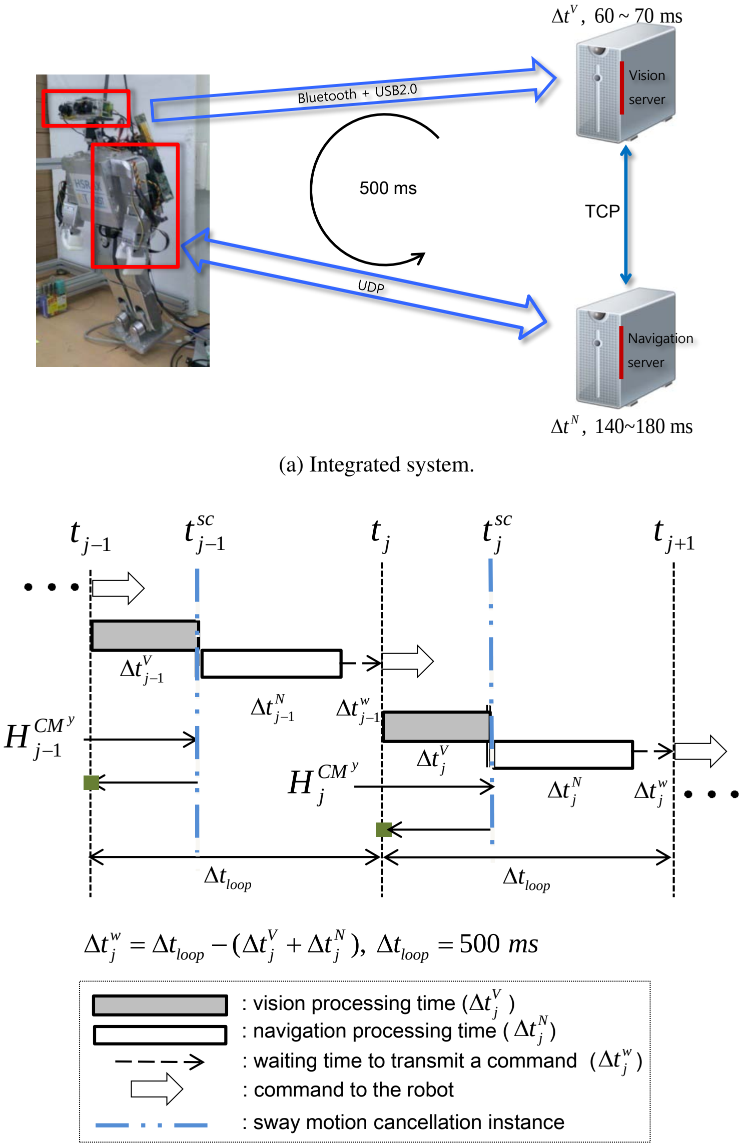
3D point $D$ along the direction of $D$ with the additional distance of $E$. In this paper, $\alpha$ and $\lambda$ are assigned as 2.0 and 5 cm, respectively.

## 3 Sway Motion Cancellation

Fig. 3 shows the system integration and communication diagrams between servers. To guarantee the constant vision processing period, vision and navigation modules are individually implemented using separate servers. The robot transmits its internal states including the CoM coordinate trajectory and supporting leg information to the navigation server at every 20 ms. This information is used for the sway motion cancellation in Section 4. In Fig. 3b, $t_j$ is the periodic instant at the $j$th control instant which the vision sensor takes the corresponding RGB-D information, $t_j^{sc}$ is the instant of the sway motion cancellation at the beginning of the navigation process, $\Delta t_j^V$, $\Delta t_j^N$, and $\Delta t_j^w$ mean vision processing, navigation and delay periods, respectively. After finishing one loop of the navigation process, the navigation server waits to transmit the corresponding walking command for $\Delta t_j^w$ to maintain the synchronization between servers.

As shown in the figure, the visual information is periodically acquired at every $\Delta t_{loop}$. The navigation server performs sway motion cancellation using $\Delta t_j^V$ at every $t_j^{sc}$. After finishing the calculation of navigation process, the navigation module calculates $\Delta t_j^w$, and waits to transmit the walking command for the corresponding control period. Through this waiting process, the navigation server can transmit its walking command at every exact control period. In this paper, $\Delta t_{loop}$ was set to 500 ms to maintain the whole processing periods consistent since the processing time periods of vision and navigation servers are not consistent. In order to obtain the exact Y coordinate of CoM of the robot at $t_j$, $CM_j^y$, the navigation server retraces the history of $CM^y$, $H_j^{CM^y}$, from the sway motion cancellation time $t_j^{sc}$ using the vision processing time $\Delta t_j^V$. $H_j^{CM^y}$ stores $CM^y$ values received from the robot through an UDP connection.

Though the sway motion-caused side effect for the obstacle detection is mainly shown along the lateral direction of the robot, Y-axis, we can find that the sway motion is also revealed in the X-coordinates of detected obstacles as a smaller oscillative pattern. In the sagittal direction, as described above, the oscillative pattern of X-coordinates for the detected obstacles is mainly caused by the compliance of legs and control inaccuracies. However, since such oscillative pattern is synchronized with $CM_j^y$ pattern, it could be also diminished by using $CM_j^y$ as for the case of lateral direction. Thus, these two directions of sway motion can be considered simultaneously. The sway motion cancellation for the sagittal direction of the robot is performed using the 2nd order polynomial-based estimation for the affect of sway motion according to the distances of obstacles from the robot along the X-axis. This is because the accuracy of 3D coordinate conversion decreases as the distances of the obstacles are increases. By measuring the peak and valley values of the coordinates of detected obstacles, the magnitude of fluctuation synchronized with the



(a) Integrated system.

(b) Communication diagram of the proposed system.

Fig. 3: Developed system environment.

sway motion from the robot has been measured. Using the measured coordinates, sway motion in sagittal direction is modeled as 2nd order polynomial as follows:

$$\bar{\mathbf{P}}^R_{o_j} = R(\theta^*_{j-1})^{-1}\hat{\mathbf{P}}^R_{o_i} - [A_{sag}\ 1]^T CM^y_j \tag{2}$$

with

$$A_{sag} = p_1\hat{x}^2_{c_i} + p_2\hat{x}_{c_i} + p_3,$$

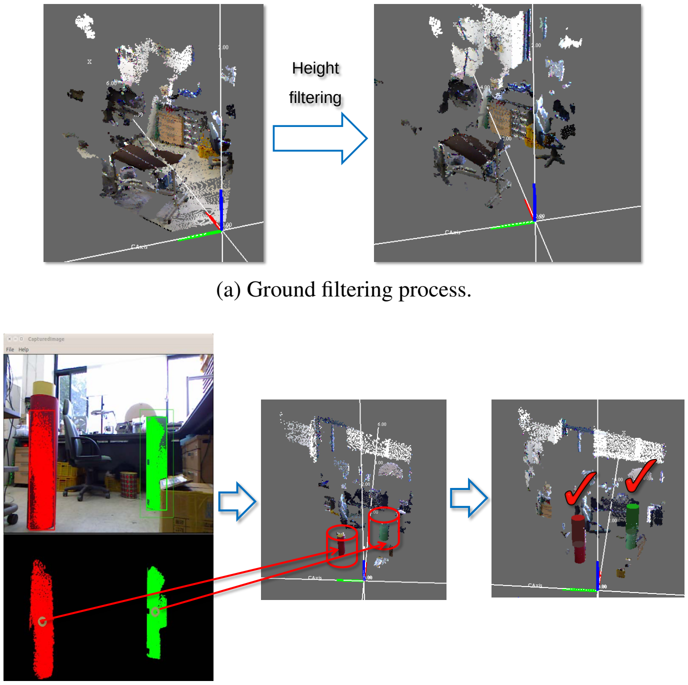
where $\hat{\mathbf{P}}^R_{o_i} = [\hat{x}_{o_j}, \hat{y}_{o_j}]^T$ is the detected coordinates of the *j*th obstacle, $\theta^*_{j-1}$ is the *j-1*th pan angle $\theta^*_{j-1}$, and $\bar{\mathbf{P}}_{o_j} = [\bar{x}_{o_j}, \bar{y}_{o_j}]^T$ is the finally obtained sway motion canceled coordinates of *i*th obstacle. $\theta^*_j = [\theta^*_{P_j}, \theta^*_{T_j}]^T$ is the pan/tilt angle for *j*th period. In this paper, the parameters for the 2nd order polynomial for (2) are assigned as $[p_1, p_2, p_3] = [0.0298, -0.0160, -0.0002]$.

## 4 Experimental Results

Fig. 4 shows the result of obstacle detection using the proposed vision system. As described in Section 2, after the ground filtering, RGB-D information-based obstacle detection was performed. In the figure, red and green obstacles were denoted using 3D cylinders according to the result of the obstacle detection process.

Fig. 5 shows the experimental setup used for sway motion cancellation experiments. As shown in the figure, two obstacles with red and green colors were used for the sway motion cancellation experiments. In order to verify the proposed sway motion cancellation scheme, sway motion cancellation experiment was performed. In the experiment, the robot walked in place along with rotating its panning angle with $20sin(\pi/33)t_i$ where $t_j$ is the *j*th time step. Note that, as the tilt angles were compensated by using $R(-\theta^*_{j-1})$ in advance which is described in Section 3, the effect of tilt angle change was not considered in this experiment. Fig. 6 shows the result of the experiment. The truncated periods of $x_{o_2}$ and $y_{o_2}$ in Fig. 6c and Fig. 6d means that the corresponding obstacle was out of sight at that periods. As shown in Fig. 6, the sway motion cancellation scheme was successfully canceled the oscillative pattern in the coordinates of obstacles generated by the sway motion of the robot during walking. As described in Section 4, the sway motion cancelated coordinates of obstacles were calculated after the compensation process of head rotation.

The performance of this sway motion cancellation scheme is also verified by comparing the mean values of the maximum variations for each of periods of sway motion canceled and the original coordinates of the two obstacles as presented in Table 1. As presented in the table, the proposed sway motion cancellation scheme was successfully canceled its affect in the coordinates of the obstacles.



(a) Ground filtering process.

(b) Obstacle detection.

Fig. 4: Obstacle detection result.

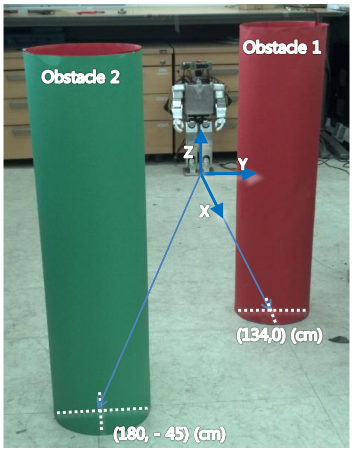
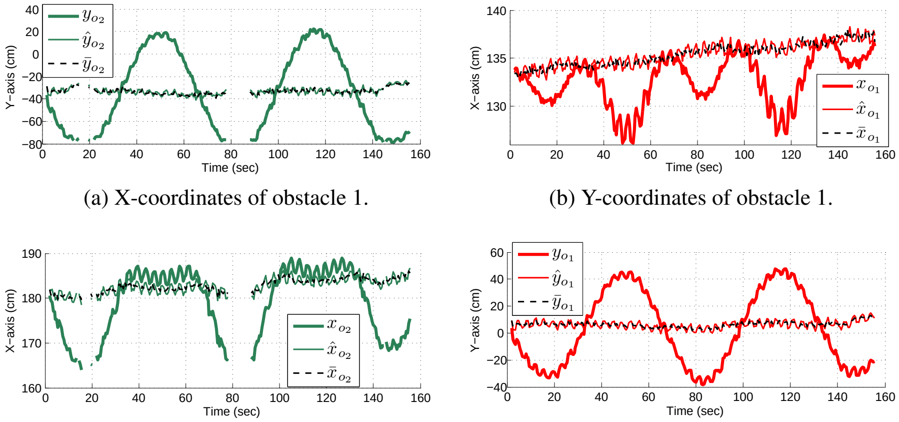


Fig. 5: Experimental setup for sway motion cancellation.



(a) X-coordinates of obstacle 1.

(b) Y-coordinates of obstacle 1.

(c) X-coordinates of obstacle 2.

(d) Y-coordinates of obstacle 2.

Fig. 6: Initial and sway motion-canceled coordinates of obstacles.

Table 1: Statistical result of the mean of the maximum variations for each of periods.

| | Coordinate | Original (cm) | Sway motion-canceled (cm) | Improvement ratio (%) |
|---|---|---|---|---|
| Obstacle 1 | *x* | 1.409 | 0.582 | 41.3 |
| | *y* | 5.976 | 2.324 | 38.8 |
| Obstacle 2 | *x* | 2.611 | 0.920 | 35.2 |
| | *y* | 5.310 | 2.841 | 53.5 |

## 5 Conclusion

This paper proposed a sway motion cancellation scheme incorporated with the CoM trajectory of the humanoid robot along with a RGB-D camera-based vision processing module. By using the depth information preprocessed by rotation using the roll and pitch angles from the attached ARS sensor, the obstacle detection and calculation of their coordinates could be performed efficiently. In particular, ground filtering process decreased the number of candidate 3D points for the calculation of obstacle coordinates, and incorporation between color and depth information was effectively considered. Using the developed vision system, sway motion cancellation scheme was applied for the coordinates of obstacles considering the coordinate of CoM. The proposed scheme was verified through experiments using HSR-IX, and showed effective sway motion cancellation performance.

## Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-0000150).

## References

1. Sakagami Y, Watanabe R, Aoyama C, Matsunaga S, Higaki N, and Fujimura K (2002) The intelligent ASIMO: system overview and integration. Paper presented at IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 2478–2483, Sept 2002
2. Akachi K, Kaneko K, Kanehira N, Ota S, Miyamori G, Hirata M, Kajita S, and Kanehiro F (2005) Development of humanoid robot HRP-3P. Paper presented at IEEE-RAS International Conference on Humanoid Robots, Tsukuba, Japan, pp 50–55, Dec 2005
3. Yoo J-K, Lee B-J, and Kim J-H (2009) Recent progress and development of the humanoid robot HanSaRam. Robot and Auton Syst 57(10):973-981. doi:10.1016/j.robot.2009.07.012
4. Lee B-J, Stonier D, Kim Y-D, Yoo J-K, and Kim J-H (2008) Modifiable walking pattern of a humanoid robot by using allowable ZMP variation. IEEE Trans Robot 24(4):917–925, doi:10.1109/TRO.2008.926859
5. Gutmann J-S, Fukuchi M, and Fujita M (2008) 3D perception and environment map generation for humanoid robot navigation. J Robot Res 27(10):1117–1134. doi:10.1177/0278364908096316
6. Okada K, Kojima M, Sagawa Y, Ichino T, Sato K and Inaba M (2006) Vision based behavior verification system of humanoid robot for daily environment tasks. Paper presented at IEEE/RAS Int Conf on Humanoid Robot, pp 7–12, Dec 2006. doi:10.1109/ICHR.2006.321356
7. Asfour T, Azad P, Vahrenkamp N, Regenstein K, Bierbaum A, Welke K, Schöder J and Dillmann R (2008) Toward humanoid manipulation in human-centred environments. Robot Auton Syst 56(1):54–65. doi:10.1016/j.robot.2007.09.013
8. Hale J G and Pollick F E (2005) 'Sticky hands': learning and generalization for cooperative physical interactions with a humanoid robot. IEEE Trans Syst Man Cybern C 35(4):512–521. doi:10.1109/TSMCC.2004.840063
9. Seara J F and Schmidt G (2004) Intelligent gaze control for vision-guided humanoid walking: methodological aspects. Robot Auton Syst 48(4):231–248. doi:10.1016/j.robot.2004.07.003
10. Yoo J-K, Kim J-H (2010) Navigation framework for humanoid robots integrating gaze control and modified-univector field method to avoid dynamic obstacles. Paper presented at IEEE/RSJ Int Conf Intell Robot and Syst, Taipei, Taiwan, pp 1683-1689, Oct 2010. doi: 10.1109/IROS.2010.5650381
11. Yoo J-K, and Kim J-H (2012) Fuzzy integra-based gaze control architecture incorporated with modified univector field navigation for humanoid robots. IEEE Trans Syst Man Cybern B Cybern 42(1):125–139. doi:10.1109/TSMCB.2011.2162234
12. Dune C, Herdt A, Stasse O, Wieber P-B, Yokoi K, and Yoshida E (2010) Cancelling the sway motion of dynamic walking in visual servoing. Paper presented at IEEE/RSJ Int Conf Intell Robot Syst, Taipei, Taiwan, pp 3175–3180, Oct 2010. doi:10.1109/IROS.2010.5649126
13. Dune C, Herdt A, Marchand E, Stasse O, Wieber P-B, and Yoshida E (2011) Vision based control for humanoid robots. Paper presented at IROS Workshop on Vis Control of Mob Robot, pp 19–26, 2011.
14. Hong Y-D, Lee B-J, and Kim J-H (2011) Command state-based modifiable walking pattern generation on an inclined plane in pitch and roll directions for humanoid robots. IEEE/ASME Trans Mech 16(4):783–789. doi: 10.1109/TMECH.2010.2089530

15. Kinect description (2011) http://en.wikipedia.org/wiki/Kinect, Accessed 15 Oct 2011
16. Disassemble process of Kinect (2011) http://www.ifixit.com/Teardown/Microsoft-Kinect-Teardown/4066/1. Accessed 15 Oct 2011
17. MyArs-USB of WithRobot (2011) http://www.withrobot.com/entry/myARS-USB. Accessed 15 Oct 2011
18. OpenNI official web site (2011) http://75.98.78.94/default.aspx. Accessed 15 Oct 2011
19. OpenCV official web site (2011) http://opencv.willowgarage.com. Accessed 17 Oct 2011
20. Nicolas Burrus (2012) Kinect calibration, http://nicolas.burrus.name/index.php/Research/Kinect. Accessed 17 Jul 2012