



Ground Vehicle Driving by Full Sized Humanoid

Kiwon Sohn¹ · Giho Jang²

Received: 18 January 2019 / Accepted: 2 December 2019 / Published online: 27 December 2019
© Springer Nature B.V. 2019

Abstract

This paper presents a technical overview of the Team DRC-Hubo@UNLV's approach to the driving task in DARPA Robotics Challenge Finals (DRC-Finals). First, hardware updates in the main body and the perception head of the contestant robot, DRC-Hubo+, were presented by comparison with the previous platform (DRC-Hubo in DRC-Trials). Then, the control system which enabled the full-sized humanoid to drive the off-the-shelf utility vehicle in the competition was provided. For this, the sensor data fusion process and the advanced driving assistant techniques were emphasized. Next, the driving pattern analysis of other type operators (such as human-drivers or pure tele-operation systems) was provided to demonstrate the performance of the developed system. Lastly, test and evaluation of the built robot and control system was provided via experimentation which DRC-Hubo+ drove the vehicle in a real-world setting. The presented approach was also verified in DRC-Finals as the author's team was placed in the top record at the driving task in the competition.

Keywords Humanoids · Vehicle driving · Darpa robotics challenge · DRC finals · DRC-Hubo+

1 Introduction

After DARPA (Defense Advanced Research Projects Agency) Grand Challenge in 2005 and Urban Challenge in 2007 [1], the development of driverless cars have been one of main focuses in the robotics research and its related industries. Google's former robot chief, Andy Rubin, has said that robots will have replaced Google's factory workers and its delivery drivers within the decade [2]. As such, autonomous robot systems which can perform given tasks and deliver products are receiving much attentions from industry now [3].

However, enabling robots to drive a vehicle is different from the driverless cars. While efforts such as Google

Car is focusing on engineering the vehicle [4], there are broader impacts and merits if robots can drive off-the-shelf vehicles. One big picture is for humanoids to drive unmodified utility vehicles and perform various tasks in human-centered workplaces. Self-driving humanoids can operate human tools and assist human workers in various stages of supply chains such as logistics, inventory and distribution. For example, the robot can load containers to a vehicle and transport them to other storage or distribution centers (DC). Such robots can assist or replace human labors in warehouse, delivery and even retail areas.

As a general purpose re-programmable machine, humanoids also has broad impacts. When needed, humanoids readily can be programmed to drive a wide variety of utility vehicles such as tractors, jeeps, trucks and even forklifts. While human workers demand the special training which take time and cost, humanoids can possibly be more quickly trained to operate specialized vehicles. It has an effect on various tasks in dangerous working environments such as mining, and fishery. Currently, humanoids rarely have the enough range-of-motion, perception and cognition to drive a vehicle autonomously. As such, endowing a humanoid with the ability to drive a vehicle has also intellectual merits.

Compared to the self-driving vehicle which has been developed for over 50 years [5], the development of humanoids which can drive off-the-shelf vehicles has not been deeply exploited. Zhang et al. [6] applied a series

This project was supported in part by a US DARPA Award #N65236-12-1-1005 for the DARPA Robotics Challenge.

✉ Kiwon Sohn
sohn@hartford.edu

Giho Jang
smile.giho@gmail.com

¹ Department of Electrical and Computer Engineering, University of Hartford, West Hartford, CT, USA

² Department of Mechanical Engineering, University of Nevada, Las Vegas, NV, USA

of low dimensional solutions to build a motion plan for simulating human-like robot models inside a vehicle. Qui et al. [7] addressed a hierarchical framework for planning and simulating humanoid's vehicle ingress motions in dynamic environments. However, driving was often limited to the control of cockpit or mock-up vehicles [8, 9] in the experimental set-up [10], not real platform driving in the previous works.

In 2012, the DARPA Robotics Challenge (DRC) [11] emphasized the impacts and merits of humanoid's vehicle driving. Motivated by Japan's Fukushima Daiichi nuclear disaster on March 2011, DARPA announced that they will kick off competitions which ask all the participant teams to develop robot systems which can replace or assist human labors in disasters. To achieve the goal, two competitions which were called DRC-Trials [12] and DRC-Finals [13] were held in 2013 and 2015 respectively. The robotics challenge demanded the robots to complete eight different tasks in a given hour under disrupted communication environments. DRC-Trials was the preliminary round match and 16 teams gathered in Florida. It allowed the participants to tele-operate robots for the vast majority of tasks. DRC-Finals was the final round match of the challenge and 25 worldwide robotics organizations competed in Pomona, California. In Finals, teams could not heavily rely on tele-operation due to the communication blackouts which were controlled by DARPA.

Vehicle driving was the first mission among eight tasks in the robotics competition (Fig. 1). DARPA explained the reason that robots should be able to drive vehicles by themselves since disasters such as radio-active area are often too toxic for human drivers. In DRC-Trials (2013), 16 world-finalists which passed qualification process were competing initially. In the first challenge, only a few robots were able to drive a given vehicle (Polaris Ranger XP) to

the goal spot. Many of the participating robots even could not start the vehicle [14]. Furthermore, the team who even finished as the first place took 5 minutes to finish the just 250 foot distance race [15].

In DRC-Trials, each team developed their own robot platforms and its tele-control systems for the unified use in all different eight tasks of the challenge [16–18]. They were built considering all the factors of driving and non-driving tasks. Accordingly, the developed platforms and systems were not often optimized for driving task [19, 20]. However, unlike other tasks which ask for robots to operate simple hand-sized tools in open-space environments (such as door-opening, valve-turning and wall-drilling), the driving task demands for robots to handle many control-inputs of the vehicle (steering wheel, gas and brake pedals). Furthermore, during the task, the robot has limited perception and range-of-motion dues to structures (poles and frames) of the vehicle. To enable the robot to finish the race within the given time limits, the human operator should make an optimized decision continuously and quickly based on the collected data from perception process.

In previous works, the authors also developed the contestant robot platform, DRC-Hubo, and its tele-control system for the competition. However, the hardware design [21] which did not fully consider the characteristics of driving task resulted in breakdown of the system during course-driving. The control system which were made for multi-task use [22, 23] also did not provide direct and easy control interface to operators. The net result is that it took almost 30 minutes to finish the driving course (in DRC-Trials).

With the lessons from DRC-Trials, the authors developed a new humanoid control system which is optimized (especially in perception data processing) for driving task. First, intuitive perception of surrounding environments is emphasized in the system. For that, related data from different sensors (such as Lidar and camera image) were merged through the fusion process. This process not only provided integrative understanding of the given driving course to human-operator but also enabled the efficient segmentation process which can distinguish the non-obstacle areas from the course. Second, unlike the previous tele-control system which relied mainly on human operator's manual control, the newly developed system adopted various advanced driver assistant functionalities. Using the kinematics model of the vehicle and sensor-measurement, the expected vehicle-path was calculated. It made the operator to estimate the vehicle's position with the control input of the moment. Among possible paths, the most optimized path was found in terms of driving distance and collision avoidance. These newly added features provided the operator with more stable and easier control of the vehicle.



Fig. 1 DRC-Hubo's driving task in DARPA Robotics Challenge Finals 2015

To ensure robustness and stability of the developed system, continuous experimentation in real outdoor environment is implemented with the updated robot platform (DRC-Hubo+). The system is also tested and verified with its performance in DRC-Finals. DRC-Hubo+ successfully completed driving a track within 55 second which is top record in the competition [24].

The paper is organized as follows: Section 2 demonstrates the hardware design changes of the robot platform. For the final design, robustness is more emphasized in both mechanical and electronic sides. Section 3 provides an overview of the control system architecture to operate DRC-Hubo+ under the controlled network. Section 4 lists several issues which were generated by the previous head and presents the design updates of DRC-Hubo+'s new sensor-head. Section 5 describes various perception data processing techniques which enabled the human operator to get more intuitive perception of surrounding environment and safe control for driving task. Section 6 shows how driving motion of DRC-Hubo+ is designed with several assistance tools. Section 7 describes experimental setup and results using the developed control system. Section 8 draws the conclusions of the paper and presents future studies.

2 Mechanical Platform Upgrade

During DRC-Trials, the original platform, DRC-Hubo,¹ demonstrated several issues in vehicle driving task. Initially, DRC-Hubo (which has the heights of 1.55 m and the weights of 60 Kg) was designed mainly for rough-terrain walking and material handling tasks (such as door opening or valve-turning) in the challenge. Therefore, the platform was not optimized to fit in the DARPA's provided vehicle (Polaris Ranger XP). Length of legs were not long enough to get inside (ingress) the vehicle by itself in safe manner. Furthermore, actuators (in wrist and arm joints) did not provide enough strong power to handle the steering-wheel of Polaris. As such, robustness in both mechanical and electronic sides is emphasized in new hardware design for DRC-Finals.

Figure 2a demonstrates DRC-Hubo+ [25] which is newly designed for driving and other manipulation tasks in DRC-Finals. As provided in the figure, the upgraded platform has the heights of 1.75 m and the weights of 80 Kg. Compared to DRC-Hubo, the heights and weights of DRC-Hubo+ are increased by 0.2 m and 20 Kg, respectively. To increase joint power, torque limits of corresponding motor and harmonic driver also become increased [26]. To reduce wrench

effects, stiffness of each mechanical part is increased. For preventing accidental contacts with environments, all external cables (which was exposed outside shells in DRC-Hubo) are removed in DRC-Hubo+. To prevent overheat of actuators, the finned air cooling and heat dissipation system is also attached to joints that need high power consumption [27].

Figure 2b presents the joint coordinate and kinematic structure of DRC-Hubo+. The robot has 32 DOF (degree of freedom). Each leg has 7 DOF including the rotational joint for the knee-wheel and each arm has 8 DOF including the finger joint. The robot's head and the body-waist joint also have 1 DOF respectively. Compared to the original platform (DRC-Hubo), though DOF of each arm is decreased by 1, the total DOF is kept same due to the wheel joint which is attached to each knee. In its head, DRC-Hubo+ is equipped with a long distance range Lidar, multi camera system, four force-torque (F/T) sensors, and a body IMU sensor. Newly, a fiber optic gyro and a head IMU sensor are attached to the hip and neck of DRC-Hubo+ respectively for more robust tracking of the vehicle (which the robot drives)'s position and rotation information.

3 Control System Architecture

The control system is also updated to reflect mechanical and electronic changes of DRC-Hubo+. As shown in Fig. 3, the control system consists of sensor, robot and operator computers. The operator computer has a sensor client and a robot client to communicate each with a sensor data server (in the sensor computer) and a robot control server (in the robot computer).

The communication between servers and clients is established by a wireless network for remote control (of DRC-Hubo+ for driving task). For utilizing the wireless network, several limitations were also introduced with consideration of the disaster scenario (which is the motivation of DRC). They include the bandwidth (300 Mbit/s) and packet size limits and data loss during communication.

To resolve the limitations, various sensor data which can generate traffic in the network was fragmented adaptively based on the packet size and bandwidth. When necessary, each data collection module in the sensor computer also converted its input data (such as raw images, Lidar, IMU and raw audio signal) to the size-reduced data (such as low resolution images, 3D points, an acceleration and a yaw angle, and mean value of raw audio) to minimize its size. The sensor server then transmitted the processed data to sensor client in the operator computer with 15 Hz of latency time.

¹DRC-Hubo (released in 2013) is the generation following the 2007 KHR-4 Hubo and 2010 Hubo+.

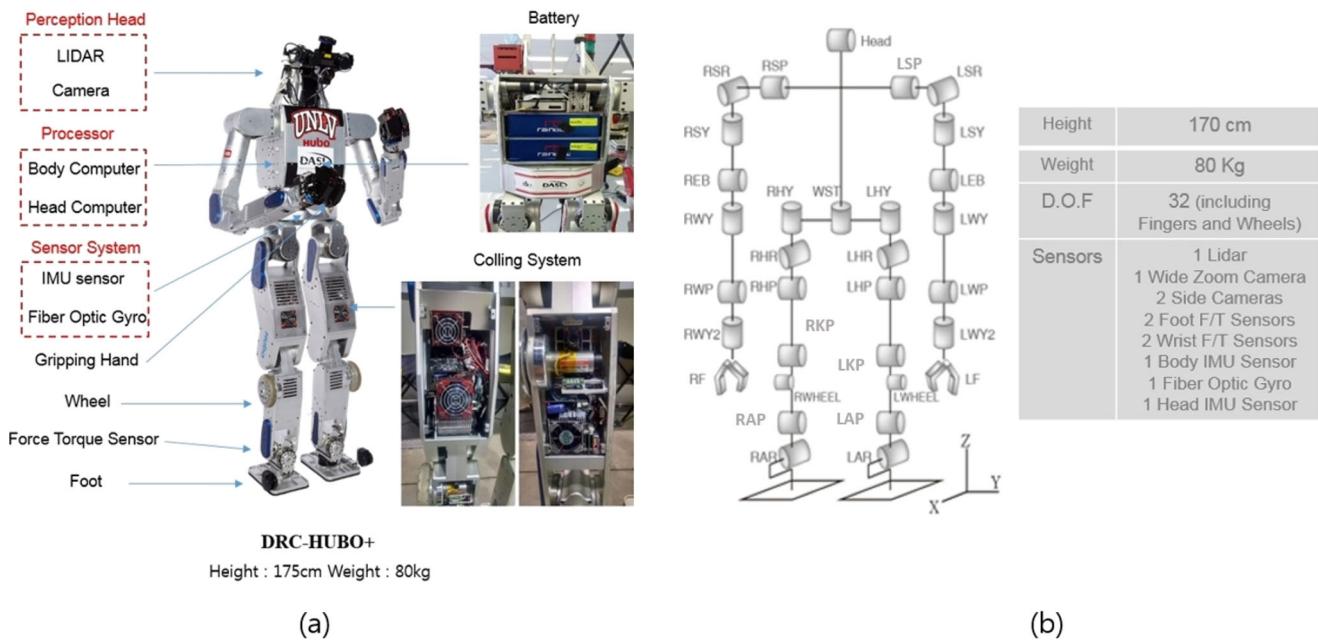


Fig. 2 a DRC-Hubo+ platform upgrade for DRC-Finals and b its joint coordinate and kinematic structure

The processed sensor data are visualized by a driving information module in the operator computer. The information module also implemented two core functionalities for humanoid's vehicle driving. They include sensor data fusion and advanced driving assistant function. These newly added

features provided the operator with more stable and easier control of the vehicle. More details with the driving assistant system is described in Section 5.

Lastly, the driving control module in the operator computer received the manipulation input (wheel angle and

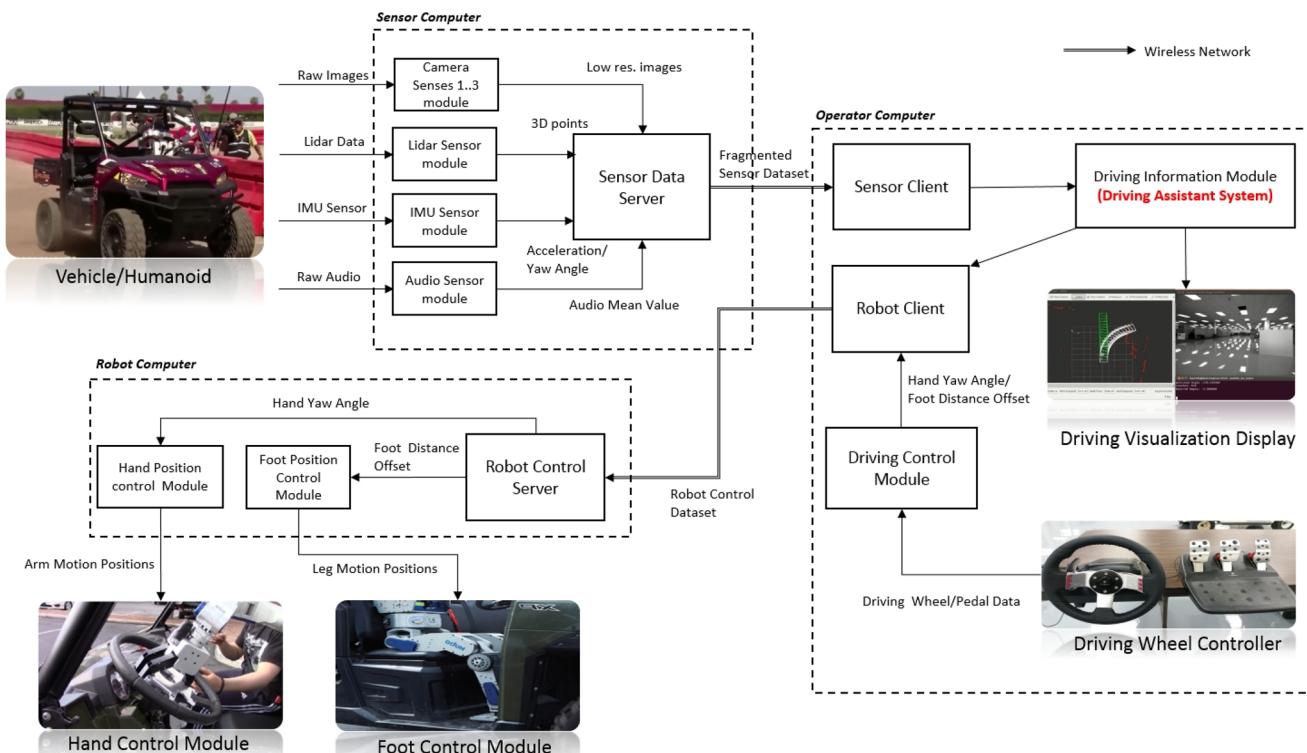


Fig. 3 Control system architecture in driving task

pedal offset) from the driving wheel controller. Combined with the output from the driving assistant system of the information module, the final control inputs (such as hand yaw and ankle pitch positions) are transferred to hand and foot position control modules (in the robot computer) respectively. The control message communication speed was also set to 15 Hz in the system.

The sensor data server and the robot control module of the architecture above are built and run in the head computer (Intel NUC i5 processor, 3M Cache 1.8 GHz) and the body computer (same specification with the head computer) respectively. The sensor data client and the driving control module are both installed in the operator computer (Dell Alienware i7 processor, 4 GHz).

4 Sensor Head Design

In DRC-Trials, original sensor head design made several issues. Since all the cameras were attached on the same base-line, only front-side view was possible to observe. Therefore, whenever the robot turn the ground-vehicle to avoid obstacles, it needed to rotate its head to track the driving course. It required extra operation time to control the pan motor which can turn the sensor head. Furthermore, rotation of the sensor head caused unexpected distortion of electric cables during the competition and it resulted in serious failure of power-supply system in the head.

With the lesson, the sensor head on DRC-Hubo+, shown in Fig. 4, is newly designed and built for DRC-Finals. It tilts with 180° without self-collision and can observe surrounding environment (180°) without pan rotation. The upgraded head has the following sensors which are all necessary for driving task:

1. Pt. Grey Flea3 camera with manual iris c-mount lens (Kowa 1/2 in. 3.5 mm F2.4): 1280 x 1024 resolution with about a 90° x 70° field of view (FOV). Used for wide-zoom color image acquisition of front-side view.
2. Hokuyo UTM-30LX-EW laser range-finder: It scans at 40 Hz over a 270° FOV at an angular resolution of 0.25°. The minimum detectable depth is 0.1 m and the maximum is 30 m. The Lidar is mounted on a dedicated tilting servo which can rotate between -90° and +90°. Aligned with the front-side view camera (in vertical) to combine image pixel values (from the camera) with intensity-like reflectance information (from the Hokuyo) for each point.
3. Two Logitech Webcam Pro 9000 cameras: 1600 x 1200 resolution with Carl Zeiss ultra-wide angle lens. Aligned with the front-side view camera, they provided DRC-Hubo+ with 180° camera-view of driving course.

4. x-IMU with 3-axis accelerometer, 3-axis magnetometer and 3-axis gyro. Data rates up to 512 Hz

Section 5 presents how the collected data (from the sensor head) be processed to provide more intuitive perception and to enable DRC-Hubo+ to drive the vehicle in a fast and safe manner.

5 Perception Data Processing in Driving Information Module

The first task in DRC-Finals was vehicle driving. In the task, contestant robots were asked to drive the utility vehicle through the outdoor driving course. Figure 5 demonstrates the ground-vehicle (Polaris Ranger XP 900) and the built driving course² which were used in the competition. For the first task, the robot began in the vehicle, drove through the course, and crossed the finish line. The task was considered complete when both rear wheels of the vehicle have crossed the finish line. As described in Section 1, the authors developed a new perception data processing system (driving information module of control system architecture in Section 3) which is optimized for driving task above.

First, intuitive perception of the given driving course is emphasized in the system. For that, related data from different sensors were merged through the data fusion process. This process provided integrative understanding of the surrounding environment to human-operator. It also enabled the efficient obstacle segmentation process which can determine the non-obstacle areas from the driving course. More details with sensor data fusion and its application are described in Section 5.1.

Second, unlike the previous tele-control approach (in DRC-Trials) which relied mainly on human operator's manual operation, the newly developed system adopted various advanced driver assistant functionalities. Thorough the kinematics modeling of the vehicle (Polaris Ranger XP 900) and sensor-measurement, the expected vehicle-path was computed. It made the operator to estimate the vehicle's position with the control input of the moment. Among all the possible vehicle-paths, the most optimized path was found in terms of driving time and collision avoidance. Section 5.2 presents how the driver assistant functionalities were developed and implemented in the new system.

5.1 Sensor Data Fusion

As presented in Section 4, the sensor head of DRC-Hubo+ has three different cameras (1 frontal view and 2 lateral

²Lewis, T., Darpa Robotics Challenge, June 6, 2015.

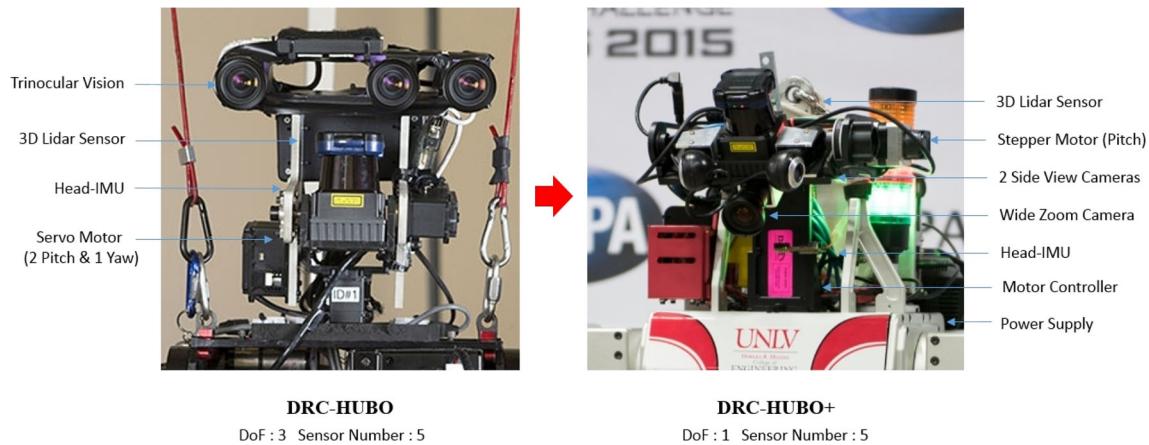


Fig. 4 Perception head of DRC-Hubo and DRC-Hubo+

side view) and 1 Lidar system (laser range finder). The frontal view camera is located on top of the Lidar system. Figure 6a demonstrate the captured image from the frontal view camera and the collected point cloud data from the Lidar system.

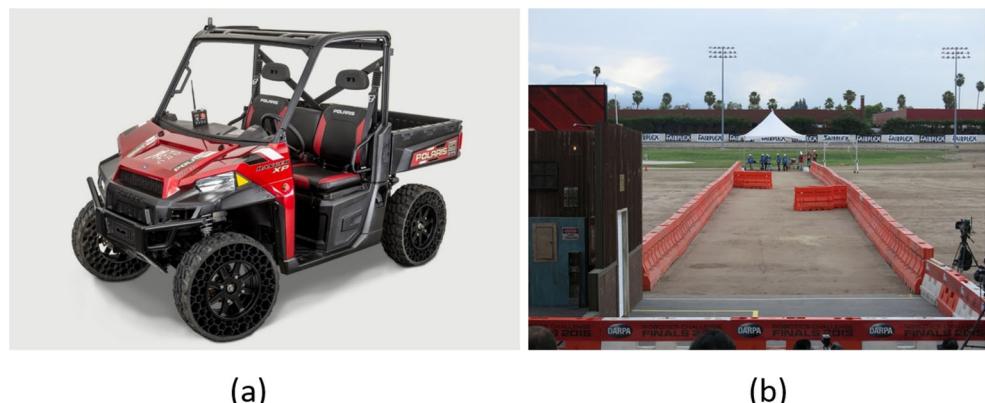
Figure 6b presents the range-image (of the collected point cloud data) which is generated through 2D visualization process. Based on the kinematics (between the frontal view camera and Lidar system) and intrinsic camera matrix data (from calibration process of the frontal view camera), the 3D point cloud data was converted to range-image. After 2D visualization, pixel matching between the original camera image and the range-image was implemented and it provided color info to the original point cloud data. Figure 6c shows the processed point cloud data. The net result is that the image and point cloud data from the two different sensors become merged through the data fusion process (which consists of 2D visualization and pixel matching).

As shown in the figure, points with no color info exist in the processed cloud data. It is because the point is out of scope in the frontal view camera perception. Use of wider-zoom camera will increase the area of points which have color info in the cloud data.

The fusion process enabled 2D image processing techniques be applicable to the collected point cloud data. As demonstrated in Figs. 7 and 8, the driving path is initially segmented from the captured camera image. For this, EM (expectation-maximization) algorithm with Gaussian mixture model is used. EM algorithm is an iterative method which can find maximum likelihood estimates (MLE) or maximum a posteriori (MAP) estimates of parameters in statistical models. It alternates between E step (which estimates the model's parameter and creates the log-likelihood probability distributions) and M step (which computes parameters which can maximize the expected log-likelihood found in the E step) [28].

Algorithm 1 presents how the method is applied for the segmentation process. There are 2 components (obstacle class when $i = 1$ and driving path class when $i = 2$) and the i 'th component is w_i . u_i and $\sigma^2 I$ are an associated mean vector and covariance matrix of the component w_i . In the initial loop, E-step first computes the probability of expected classes (using pre-determined mean values of two classes) for each pixel in the image. With the calculated probability, M-step computes the maximum likelihood mean value of each class. The estimated mean values are then used in E-step of the second

Fig. 5 a Polaris Ranger XP and b driving course



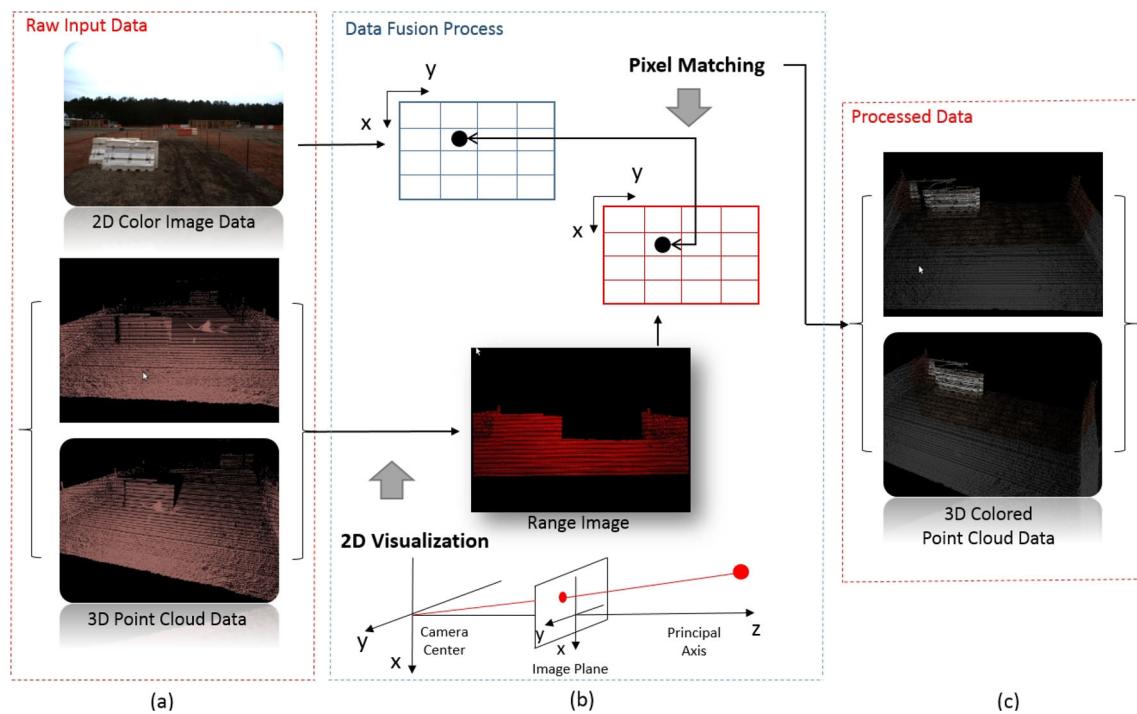


Fig. 6 **a** Raw data from sensors: captured image from frontal view camera (top) and 3D Point cloud data from Lidar (bottom), **b** 2D visualized range-image and **c** Processed Point cloud data

Fig. 7 Driving Course
Segmentation: **a** user-aided
obstacle candidates assignment
and **b** univariate Gaussian
distribution case

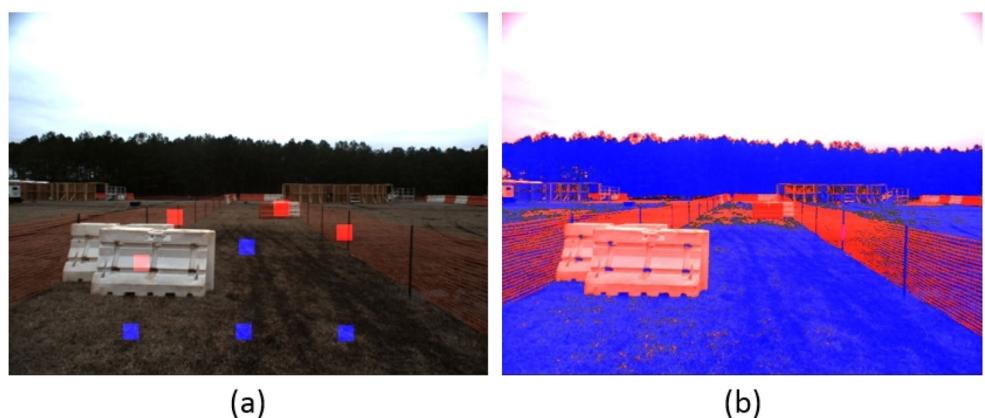
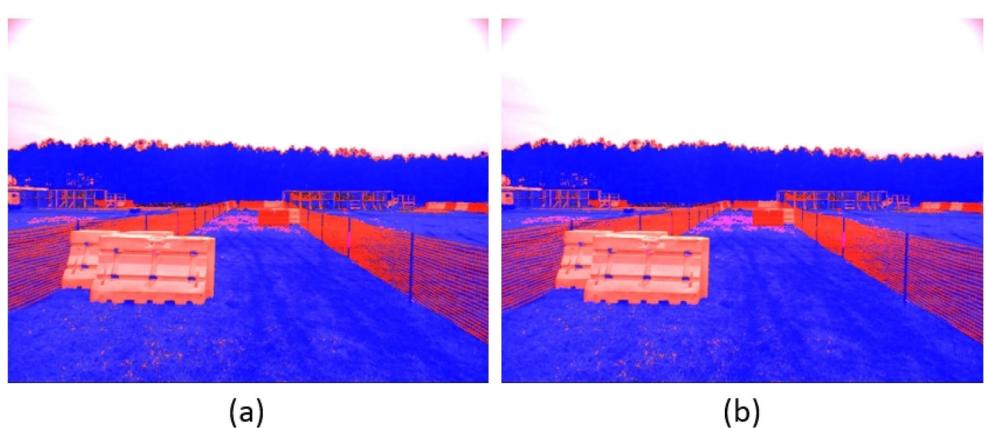


Fig. 8 Multivariate Gaussian
distribution cases: **a** dimension
= 8 and **b** dimension = 9



loop. Until the mean values become converged, E-step and M-step are iterated continuously.

Figure 7a presents the initially selected samples of each driving path class (blue) and obstacle class (red) in the scene. Figure 7b shows the output of segmentation process (when one-dimensional normal distribution is used) with the initially classified data.

Algorithm 1 EM for Gaussian mixture model in driving path segmentation.

Iterate until Convergence ($\lambda_t = u_1(t), u_2(t)$ on the t'th iteration).

E-step

Compute expected classes of each datapoint (x_k which is each pixel) for each class: Obstacle (i=1) and

Driving path (i=2)

$$P(w_i|x_k, \lambda_t) = \frac{p(x_k|w_i, \lambda_t)*P(w_i|\lambda_t)}{p(x_k|\lambda_t)}$$

$$= \frac{p(x_k|w_i, u_i(t), \sigma^2 I)*p_i(t)}{\sum_{j=1}^2 p(x_k|w_j, u_j(t), \sigma^2 I)*p_j(t)} \text{ M-step}$$

Computer maximum likelihood value of u given data's class membership distributions

$$i_i(t+1) = \frac{\sum_k P(w_i|x_k, \lambda_t)*x_k}{\sum_k P(w_i|x_k, \lambda_t)}$$

Figure 8 presents results when multivariate normal distributions are used (8 and 15 respectively). Each dimension correspond to the number of selected samples in initial classification. Compared to the univariate case, multivariate ones present clear segmentation results. However, for most scenes (of driving course), when the dimension is bigger than 8, the performance did not change much. Therefore, samples between 8 and 10 were selected in the initial classification process.

The path segmented image above (generated from EM algorithm using the original image which is captured by 2D camera) is combined with the 3D point cloud data (which is collected by the lidar system in DRC-Hubo+'s sensor head) as demonstrated in Fig. 6. Algorithm 2 presents more details with the process. In the algorithm, f_x and f_y denote focal lengths of the 2D camera and c_x and c_y reflect the kinematic offsets between the 2D camera and the lidar system.

Through the fusion process above, the driving path is divided from the scene in point cloud data. Figure 9a and b shows the segmentation result in the cloud data. Initially, the segmentation process is implemented only in the points which are inside the scope of the frontal camera perception. Therefore, using the selected set of the processed points, the kinematic configuration of the driving path (slope and height variance) is calculated through the regression process. The estimated kinematic data expanded the segmentation process for whole area of the point cloud

Algorithm 2 Fusion process between driving path segmented image and point cloud data.

Input: Driving path segmented image from EM algorithm and original 3D point cloud data

Output: Point cloud data combined with path segmentation information

for all detected points in collected cloud data, **do**

find 3D position (x, y and z) and calculate x' and y':
 $x' = x/z$ and $y' = y/z$

calculate u and v (estimated pixel location in the point cloud's 2D visualized image) and save the values:

$u = f_x * x' + c_x$ and $v = f_y * y' + c_y$

for all u, v in the point cloud image, **do**

compare the nearest pixel in the path segmented image and determine whether it belongs to obstacle or path
 go back to the corresponding point of u and v in original 3D point cloud data and save the path information which is found from the nearest pixel comparison step above

data. Figure 9c and d shows the final segmentation result (of the point cloud data) in top and canonical view respectively.

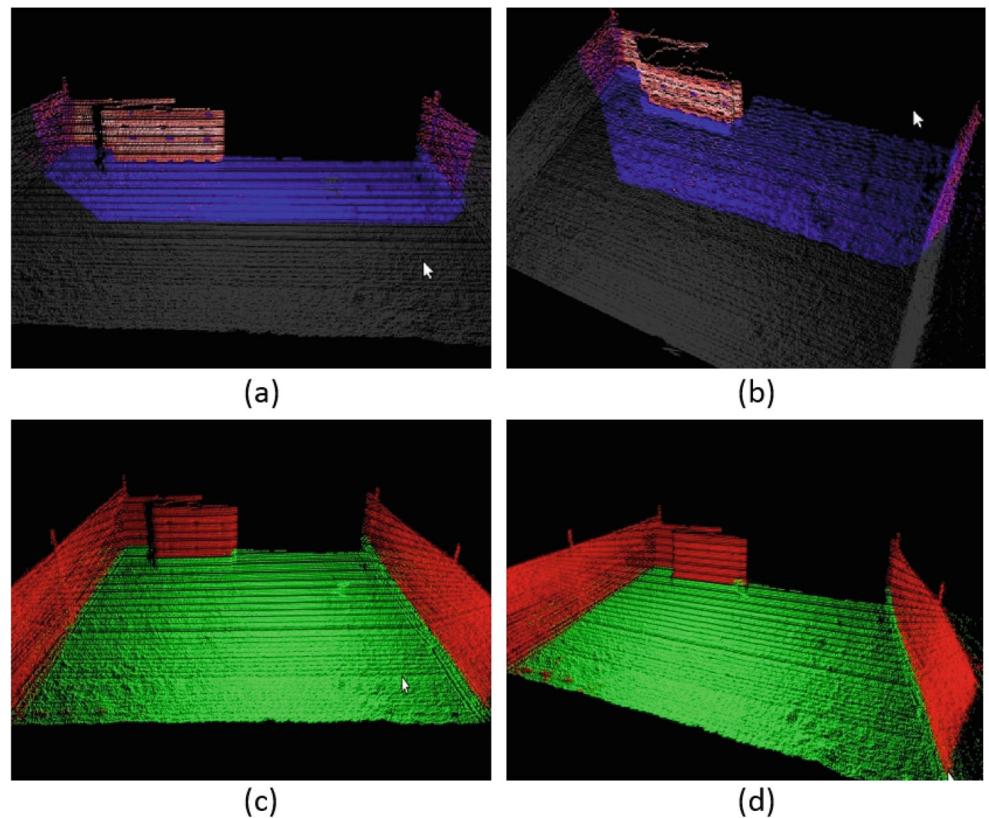
The Lidar (Hokuyo UTM-30LX-EW) which is used requires tilting rotation to generate the 3D point cloud data. This process takes one to several seconds depending the desired resolution. While driving, several second scanning was not the issue when the vehicle stopped its moving (such as stop and wait for steering-wheel turning). However, when it is moving, such time-taking scanning process can corrupt accuracy of the collected cloud data. Therefore, the Lidar was fixed at previously-defined tilting angle (10 degree towards terrain in this study) with no rotation when the vehicle is moving. It resulted in acquisition of 2D point cloud data as shown in left figure of Fig. 10.

The fusion and segmentation process which were used for 3D point cloud data above are applied to the 2D data in the same manner. Middle figure in Fig. 10 demonstrates the divided non-obstacles (green) and obstacles (red) in the scene. Right figure shows the only obstacles of the 2D point cloud data. The net result is that driving-path segmentation process is successfully implemented both in 2D and 3D point cloud data for safe driving of the vehicle.

5.2 Driving Assistant

The segmented 2D point cloud data is displayed in the user-control interface (of driving information module) with the captured images from each camera sensor (Fig. 11). Obstacles of the driving course is indicated as red points in the cloud data display. Estimated path of the vehicle is also demonstrated in the display as a series of white box models.

Fig. 9 Data Fusion between the processed image and point cloud: **a** top view and **b** canonical view; course-segmented cloud data: **c** top view and **d** canonical view



The model is a bounding box of Polaris which DRC-Hubo+ drives and is projected on the 2D point cloud data space. The estimated path is calculated based on forward kinematics modeling of the vehicle [29]. For the kinematic analysis, 1) configuration information of Polaris, 2) speed of the vehicle, 3) total time duration of the estimated path and 4) steering wheel angle at the moment were used. Figure 11 shows the displayed data in three different control-input cases (steering wheel angles of Polaris in this study). When the estimated path (bounding box series) become collapsed with obstacles in the scene, the obstacle points which are inside the path are counted and indicated as different color in the display.

Beyond path estimation, the optimal driving path of the scene is also computed and presented in the display of the

user-control interface. Figure 12 shows the computed path (a series of green boxes) in the 2D point cloud data. The optimal path is calculated by finding the steering wheel angle which can minimize collisions with obstacles in the scene while vehicle is orienting to goal spot of the driving path.

To implement it, for each possible steering angle (which range is inside the mechanical constraint of the wheel), the corresponding candidate driving path is estimated first. Then, the numerical iteration which finds the angle that minimizes two penalty values are implemented (1) based on the greedy search algorithm [30]. The first penalty value is earned by counting the obstacle point which collapse with the candidate path. The second penalty value is calculated by displacement between the goal-spot of the given driving

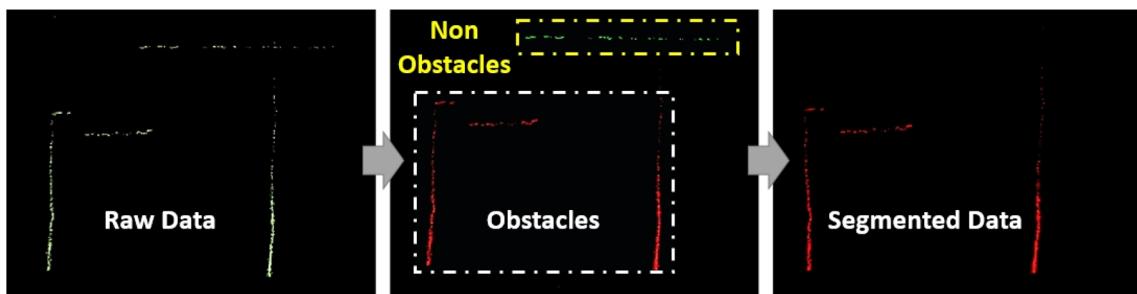


Fig. 10 Raw data from Lidar (left), course-segmented point cloud (middle and right) in 2D space

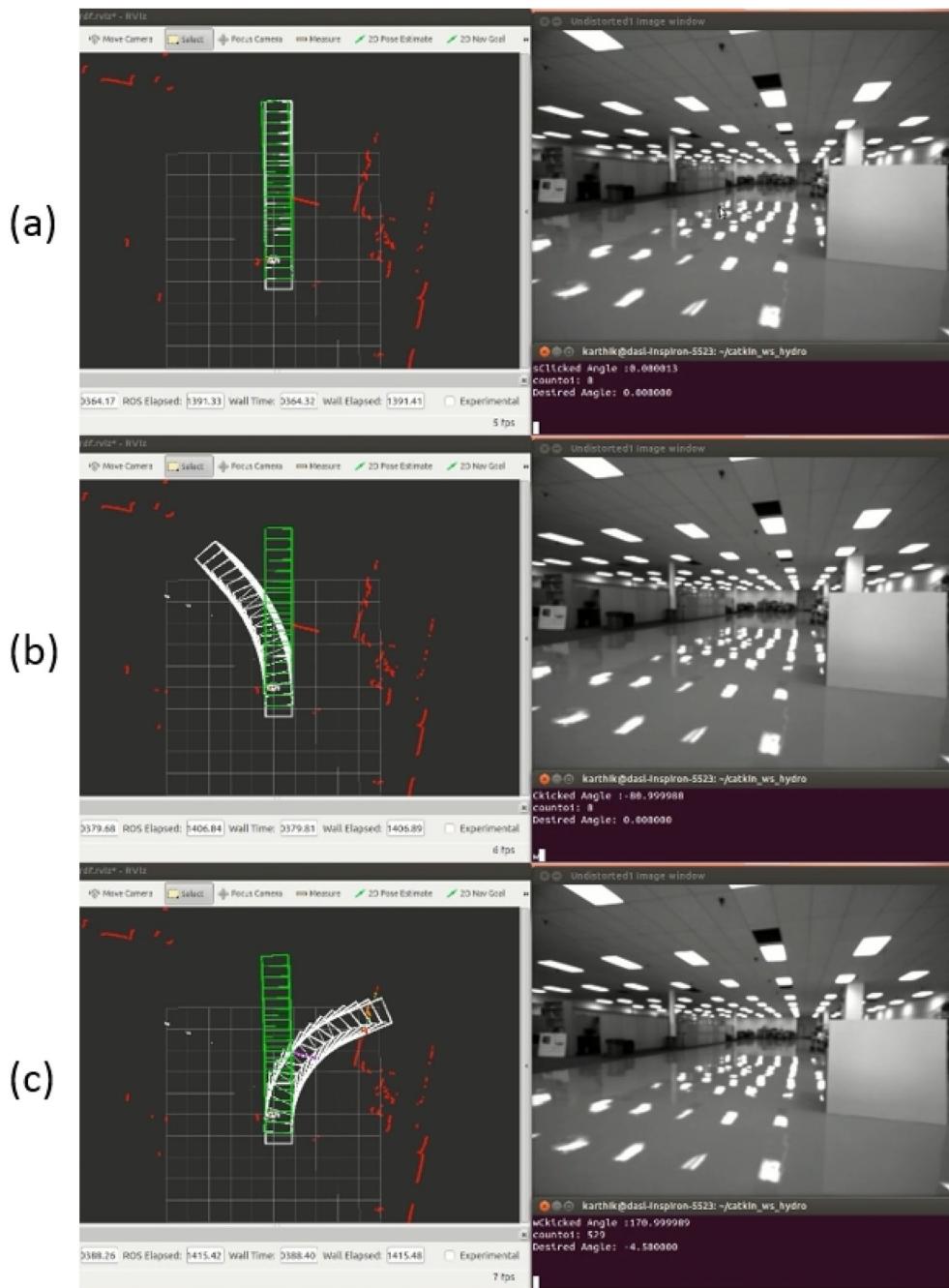


Fig. 11 **a** Path estimation and collision-detection when straight-moving, **b** left-turn and **c** right-turn

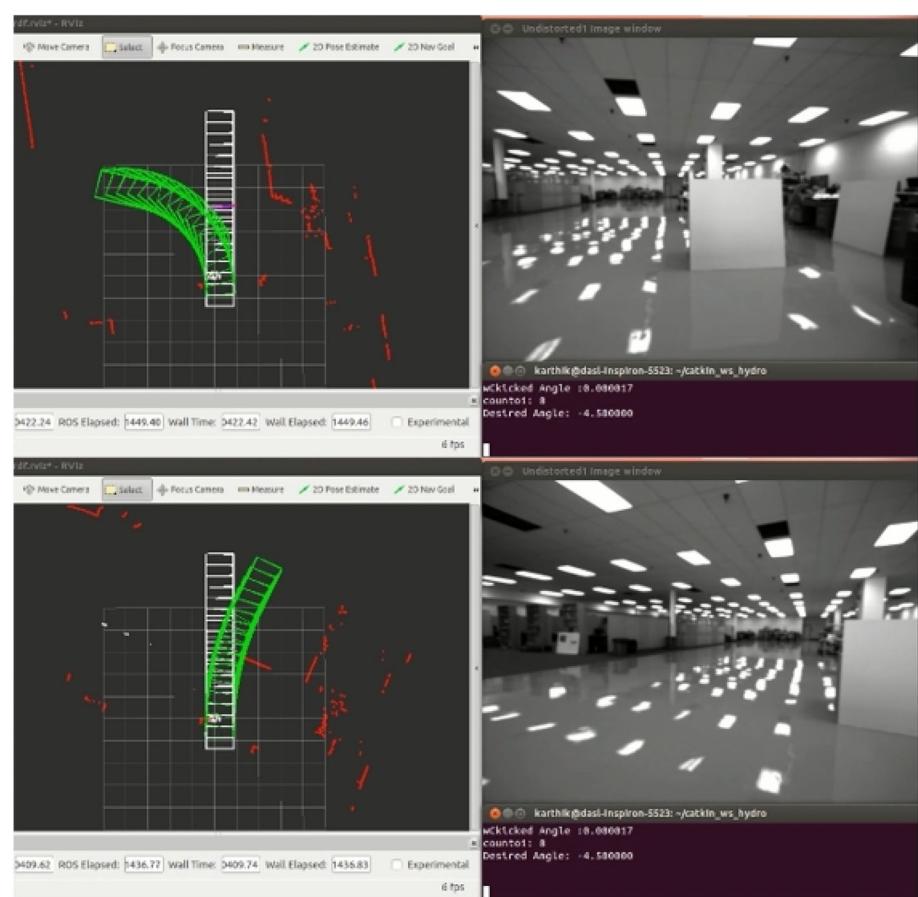
course and the end-point of the candidate path. The net result is that the candidate path which has the minimum penalty value can be assumed as the path which has lowest level of collision and shortest driving path.

$$\arg \min_{\theta} (\alpha * f(\theta, v, t) + \beta * g(\theta, v, t)) \quad (1)$$

where θ is the candidate steering wheel rotation angle, v is the velocity of the driving vehicle and t is the total time duration of the computed path. $f(\theta, v, t)$ and $g(\theta, v, t)$ are

penalty value functions which present collision-level and displacement from goal-spot respectively. While v and t are determined with the operator's choice, θ is generated from the iteration process above. In reinforcement learning, different weights on the cost value functions can result in diverse output trajectories. Therefore, the weighting factor value (α and β) on each penalty function should be chosen considering the relative importance of the corresponding penalty. In this study, 0.5 was assigned for both factors and this made the optimal path estimation process above

Fig. 12 The optimized path calculation



has same dependency on collision avoidance and goal convergence.

The computed path was used for guideline of human operator's driving. During the driving task, the operator just needed to align the estimated path (white box) to the optimal path (green box) in the user-control interface. Figure 13 shows how the driving assistant system is activated in the interface. For this experimentation, the mock-up of the DRC's driving course is built inside a warehouse. Figure 13 also demonstrates how the panorama image (which is generated from the frontal camera and two side cameras of the new sensor head design) is presented in the data display window of the user-interface.

First, Fig. 13a shows the beginning of the driving course. Since the barrier is not in the vehicle's estimated path, the optimal driving path (green box) just recommended the direction which is going toward the goal spot. The driving assistant system makes similar advice (path to the direction of the goal) even when the vehicle is in the mid-way if there are no obstacles in its estimated path.

As the vehicle moves closer to the barrier on the right direction, the optimal driving path (green box) recommended left turn (see Fig. 13b). The operator followed the direction and rotated steering wheel until the

estimated path (white box) become aligned with the optimal path (see Fig. 13c). While the vehicle is moving forward, the operator kept the alignment and the vehicle passed the barrier with no-collisions.

After passing the barrier on the right side, the optimal driving path (green box) recommended right-turn (see Fig. 13e). It is because the goal spot of the given course is located on the right side of the vehicle at the moment. Again, the operator followed the direction and rotated steering wheel until the estimated path (white box) become aligned with the optimal path (see Fig. 13f). While the vehicle is moving forward, the operator kept the alignment. It made the vehicle became oriented to the goal-line of the given course.

The optimal path (demonstrated in Fig. 13) is determined based on Eq. 1 as explained above. Among candidate paths, the driving assistant system chose the path with the minimum value of the equation which is computed by counting the obstacle pixels which collapse with the candidate path and by measuring the pixel-displacement between the goal-spot and the end-point of the candidate path.

In sum, the newly built perception data processing system provided the estimated vehicle-path and it made the

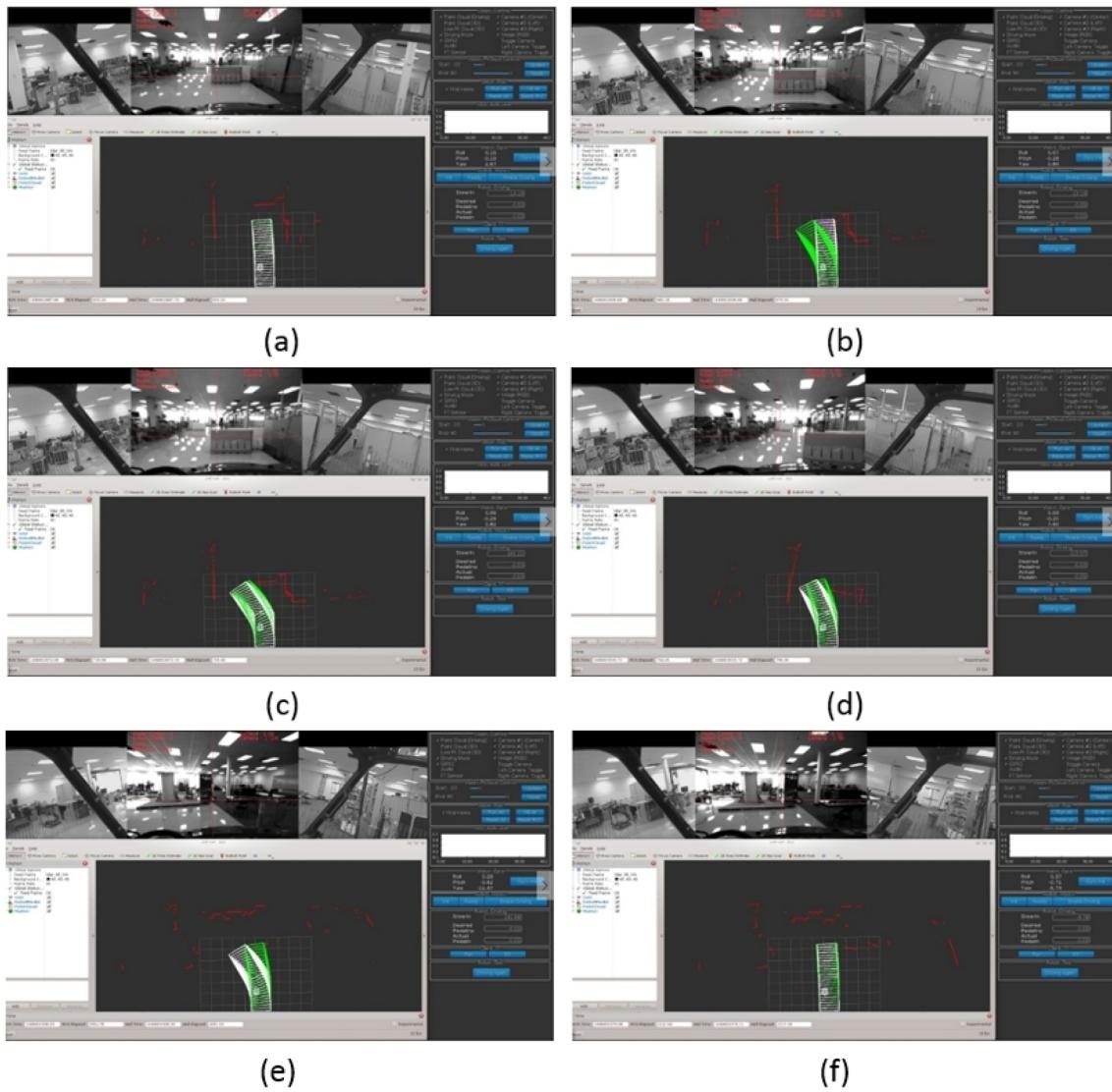


Fig. 13 Driving assistant system activation in the mock-up course

operator to predict the vehicle's position with the control input of the moment. Then, the most optimized path was also computed in terms of collision avoidance and driving distance and displayed in the user-control interface. These features enabled the operator to control the vehicle with easier and more stable manner.

6 Driving Motion

As described in Section 5, the control input of the driving assistant system is the rotated angle of steering wheel. In the previous system (which was used for DRC-Trials), the authors typed the rotation angle manually using a keyboard. However, such hard-coding resulted in significant delays in vehicle control. In this study, Logitech G27 racing wheel

system (which consists of steering wheel and pedal) is adopted (see Fig. 14).

As human operator manipulates control inputs of the system, DRC-Hubo+ made the corresponding motions (as presented in control system architecture in Section 3). Rotation of the steering wheel in the racing system resulted in movements of wrist-yaw joint of DRC-Hubo+ (hand position control module in Fig. 3). Pushing the gas pedal also made the rotation of ankle pitch joint (foot position control module in Fig. 3). Such controlled motion of DRC-Hubo+ made the real control of Polaris. Considering kinematic discrepancy between end-effector of DRC-Hubo+ and controls (steering wheel with 0.35 m diameter and pedal with 0.09 m height & 0.05 m width) of Polaris, driving aid devices were designed and added to the vehicle as shown in Fig. 15. The vehicle's original steering

Fig. 14 Control input system for vehicle driving task



wheel has a rounded center hub which DRC-Hubo+'s hand can not grab firmly. Since the vehicle's throttle foot pedal has curved design, DRC-Hubo+'s foot also can not provide stable contact to the control. To solve the issues, the steering aid which has straight planes is placed on top of the vehicle's steering wheel and the pedal aid which has an pushing assistant component is placed on top of the vehicle ground. As DRC-Hubo+ pushes the "pedal-assistant" component of the pedal pushing aid, the "pedal-pusher" component is triggered and pushed the gas pedal of Polaris. It was designed to facilitate egress motion of DRC-Hubo+ after driving task.

Figure 16 demonstrates DRC-Hubo+ with the driving aid devices in Polaris. Though legs of DRC-Hubo+ were

located outside of the sitting area, it controlled the pedal of Polaris successfully with benefits of the driving aid devices.

7 Experimental Results

Before experimenting with DRC-Hubo+ and the developed driving system, some useful data were collected from human drivers and pure-tele-operated driving system.

First, the authors observed driving pattern of human drivers with different levels. The driving level was determined based on the possessed license types and driving period. The collected measurement demonstrated meaningful differences among drivers. Figure 17 shows the

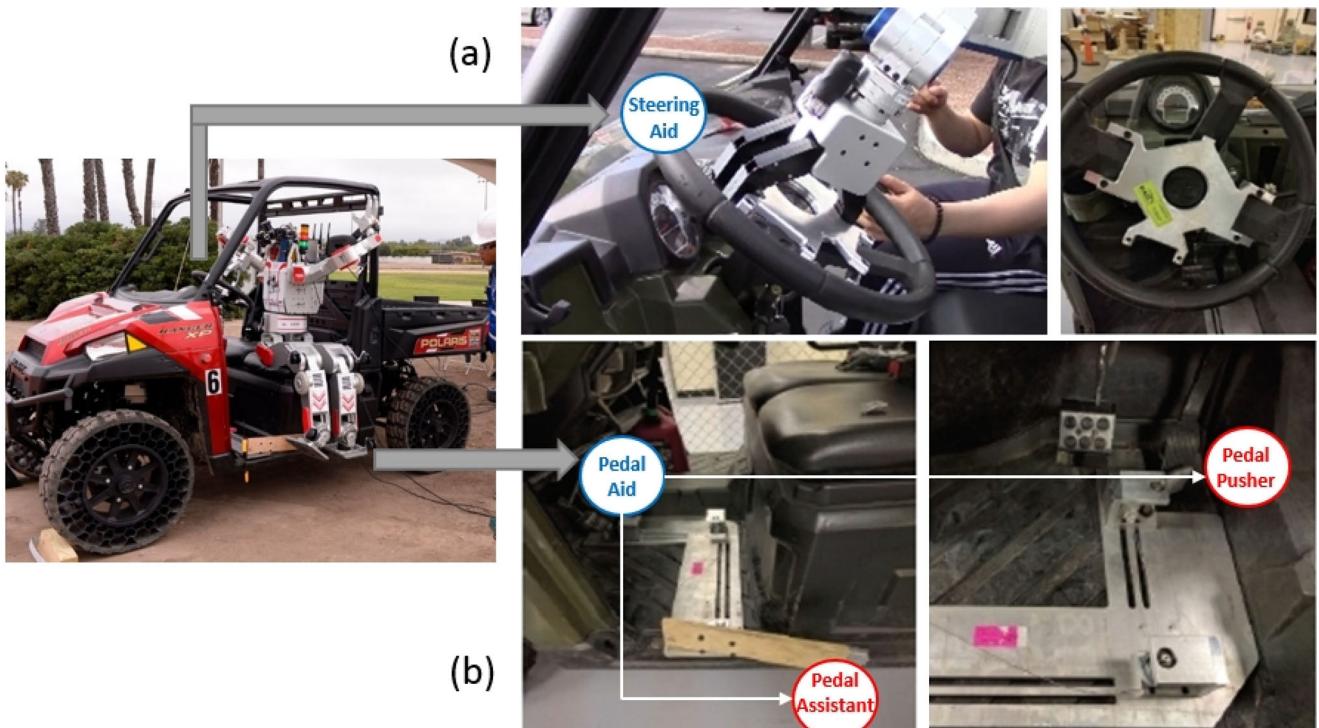


Fig. 15 Driving motion aiding devices: **a** steering aid which is designed to facilitate for DRC-Hubo+ to grab the steering-wheel and **b** pedal pusher which is wire-triggered by the pedal assistant when DRC-Hubo+ pushes down

Fig. 16 DRC-Hubo+ in Polaris

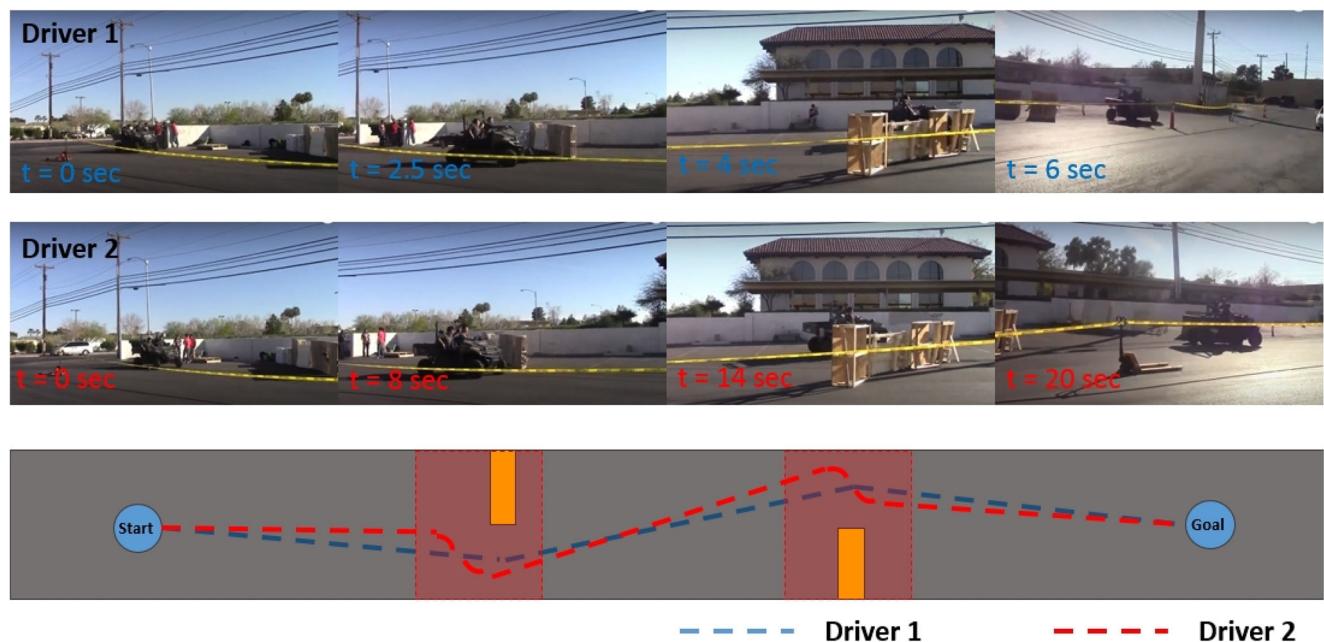
comparison between the experienced driver (Driver 1 who had served as an unmanned system pilot and trainer for 14 years in U.S. Army) and the beginner level one (Driver 2 who was a new driver with age 24). As demonstrated in the figure, the non-experienced drivers tended to stop and turn the vehicle's heading direction just in front of obstacles. It resulted in significant delay (driving time) and their measured average time record (to finish the course) was about 21–32 second. On the other hand, the experienced driver gradually turn the heading direction while the vehicle is approaching obstacles (predictive driving skill). It resulted in continuous and smooth driving path and reduced their driving time (measured at 7–11 second). For this experimentation, mock-up of DRC driving course is built in outdoor parking lot (130 meter length).

Next, the authors also observed and recorded when the pure tele-operated driving system is used. For this, KAIRO system is adopted. Figure 18 shows driving with the KAIRO

system in the driving course of DRC Finals mock-up site (on March 2015, DARPA built the competition mock-up site in Charleston, South Carolina and provided access to the finalists). The measured average time record (to finish the course) was about 58–62 second with the KAIRO system. The KAIRO system only provided real-time 2D camera image feedback to the operator. Therefore, the human operator had difficulties with making rapid decision while avoiding collision with obstacles. Furthermore, since the system was solely based on the decision of human operator, achieving more cut in driving time was difficult.

The collected data above (from experiments with human driver and the pure tele-operation system) provided the lesson that the driving assistant system for predictive driving is essential for reduction in driving time.

To test driving performance, DRC-Hubo+ was initially controlled through only tele-operation control (with no driving assistant functionality) similar to the KAIRO

**Fig. 17** Experienced and beginner level driver

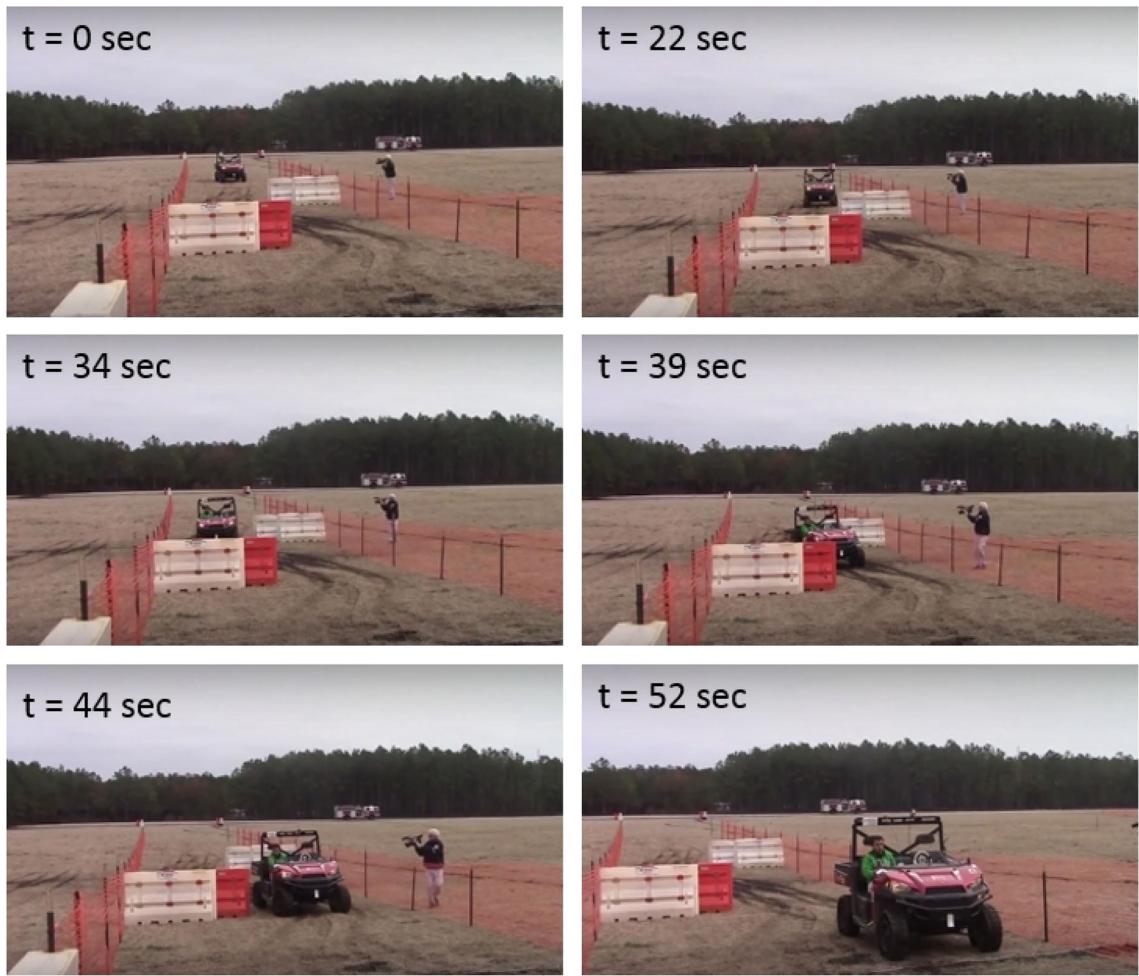


Fig. 18 KAIRO system's driving

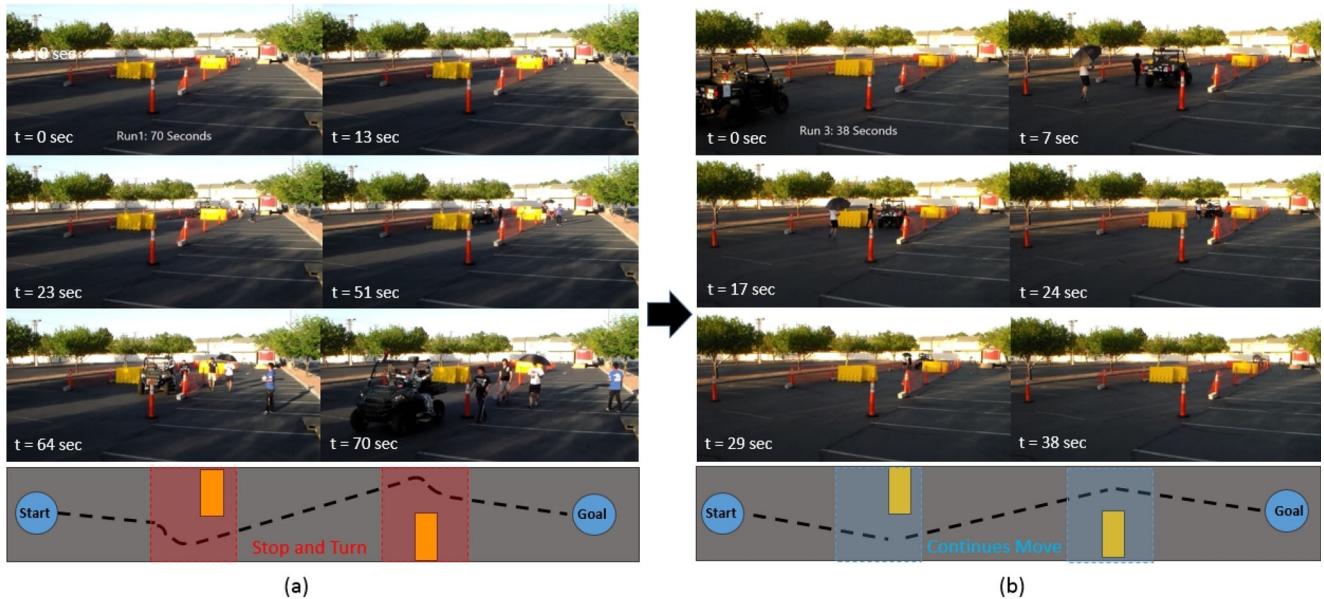


Fig. 19 DRC-Hubo+'s driving in the mock-up course

system. In the effort, DRC-Hubo+ took 58 to 270 seconds to finish the task. Figure 19a shows the DRC-Hubo+ driving in the mock-up course.

After combining the driving assistant system and having enough training for the operator, the record became much faster. Figure 19b shows the DRC-Hubo+ driving with the full system in the mock-up course. The record was measured between 29 - 55 second through multiple runs. The interesting fact is that DRC-Hubo+ demonstrated the driving pattern similar to the experienced human driver's one. With the benefits of driving assistance, DRC-Hubo+ proactively manipulated control-input of the vehicle while it was approaching obstacles. It resulted in reduction of driving time as well as robust collision avoidance. Figure 20 demonstrates the system's activation in the mock-up course for vehicle driving task.

When 5 minutes were assigned to the time limit of the driving task completion (based on the rule of DRC-Finals which asks the completion of 8 tasks within 60 minutes including penalty time), the averaged success rate of the task was slightly over than 92 percent from the measured tests. The time-record measurement of vehicle driving task is carried out by calculating the time difference between the “driving-start-point” (which Polaris started to move) and the “driving-end-point” (which rear tires of Polaris passed the

Table 1 Success rate and time range of 5 selected tasks [25]

Tasks	Success rate [%]	Elapsed time range [s]
Driving	92.1	29 - 122 29 - 55 with driving assistant system
Door	94.4	43 - 95
Valve	93.0	55 - 129
Rubble	86.3	30 - 228
Stairs	91.2	218 - 298
Avg. Rate	91.4	

goal line). To keep consistent performance for driving task, the same operator controlled DRC-Hubo+ during the team's DRC preparation period. To minimize mistakes which can be caused by improvised decisions, the operator was trained to mainly follow the advised commands from the driving assistant system (which is described in Section 5.2). Table 1 presents the success rate and time range of 5 selected tasks including vehicle driving.

DRC-Hubo+ and the developed driving system verified its performance in real-world setting. In DRC-Finals, DRC-Hubo+ successfully drove the driving course which DARPA built and provided (Fig. 21). The record was 55 second

Fig. 20 DRC-Hubo+'s driving assistant system in the mock-up course

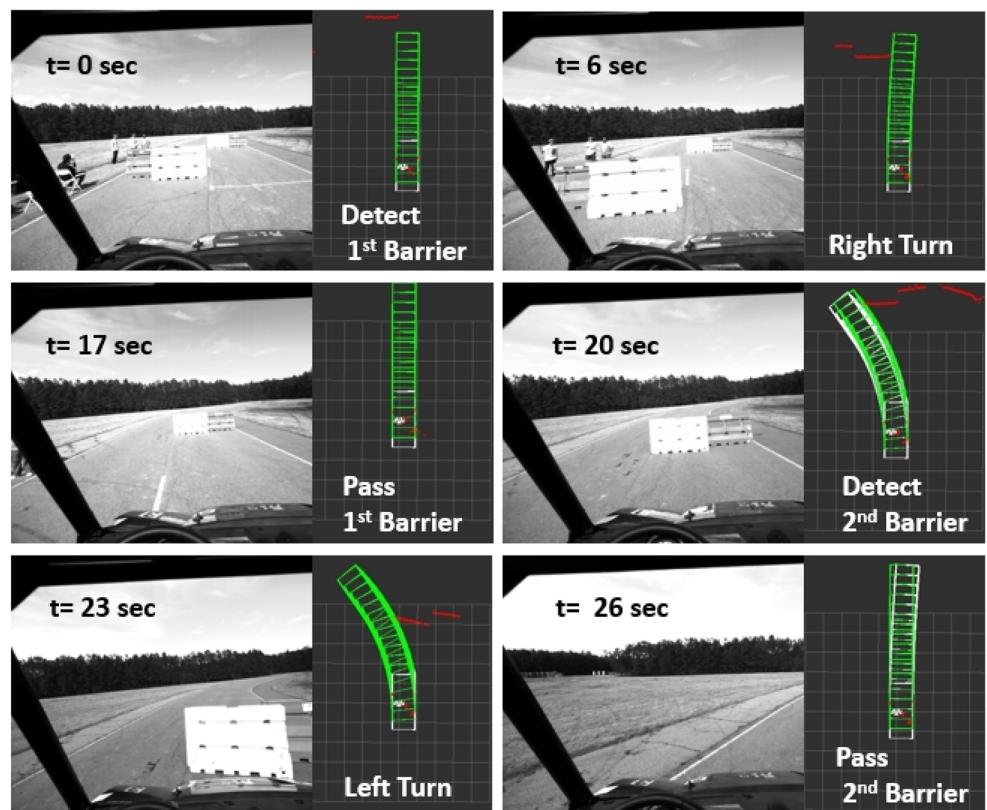


Fig. 21 DRC-Hubo+'s driving (with full system)



and was placed in the first place against other world-wide finalists. Table 2 shows the record of best performance team and the competition average in 5 selected tasks. The presented approach of Team DRC-Hubo@UNLV is also broad-casted in PBS' NOVA program on February 24th, 2016 [31].

Currently, similar efforts with the kinematically reconfigurable humanoid is under study [32]. DRC-Hubo+ which its hardware design focused mainly on the given vehicle platform can not efficiently manipulate different kinds of vehicles dues to its kinematic limits. To resolve the issue,

Table 2 Record of Each Task in DRC-Finals 2015 [24] (S denotes the number of teams which succeed the task and F denotes the number of teams which failed the task)

Tasks	S	F	Best team and record	Average
Driving	19	4	DRC-Hubo@UNLV: 00:55	02:12
Door	18	5	NEDO-JSK: 00:50	02:08
Valve	16	7	KAIST: 00:33	01:46
Rubble	8	15	DRC-Hubo@UNLV: 00:57	03:00
Stairs	7	16	NEDO-JSK: 02:25	04:39

each limb of the new platform's lower body is designed to be manually adjustable. As such, the kinematic features of the robot can be optimized depending on the given vehicle's structure.

8 Conclusion

In this paper, a technical overview of Team DRC-Hubo's approach to the driving task in DRC-Finals is presented. First, updates in both hardware and software aspects of DRC-Hubo+ were described compared to the its previous platform. Then, the control system architecture which consists of sensor, robot and operator computers was presented. Especially, the sensor data fusion process and the advanced driving assistant techniques (core functionalities of the driving information module in the system) were emphasized in this paper. Next, the performance analysis of human drivers and KAIRO system was provided to demonstrate the importance of predictive driving. Lastly, test-evaluation and verification of the developed system were provided through experimentation which DRC-Hubo+ drove the vehicle in the mock-up and the official course

of DRC-Finals. As shown in this study, DRC-Hubo+'s mechanical design is mainly focused on the Polaros Ranger XP which DARPA chosen for the competition. Due to the limitations, it could not efficiently manipulate other types of vehicles. Currently, the new platform which can optimize its kinematic adaptability depending on the assigned vehicle is under development.

Acknowledgments The authors gratefully acknowledge the contribution of Paul Oh, Youngbum Jun, Donghyun Ahn, Baek-Kyu Cho, KAIST-Hubo Lab, Juseong Shin, Kwangwoo Lee, Pareshkumar Brahmbhatt, Praxis Aerospace, University of Delaware, Ohio State University, Swarthmore College, Georgia Tech, Purdue University, Columbia University, Worcester Polytechnic Institute, Indiana University and Drexel University.

References

1. DARPA: DARPA grand challenge. Retrieved October, from <http://archive.darpa.mil/grandchallenge/> (2012)
2. Leonard, T.: How Google's self-driving car works. Retrieved February, from iol scitech (2013)
3. Clalautti, P.: Top 10 jobs that might not survive the coming robot 'jobocalypse', is yours on the list? Retrieved January, from Design and Trend (2014)
4. Tsugawa, S.: Vision-based vehicles in Japan: machine vision systems and driving control systems. *IEEE Trans. Industr. Electron.* **41**(4), 398–405 (1994)
5. Hongo, T., Arakawa, H., Sugimoto, G., Tange, K., Yamamoto, Y.: An automatic guidance system of a self-controlled vehicle. *IEEE Trans. Industr. Electron.* **34**(1), 5–10 (1987)
6. Zhang, L., Pan, J., ManocHa, D.: Motion planning of human like robots using constrained coordination. In: IEEE International Conference on Humanoid Robots, pp. 188–195 (2009)
7. Qui, Z., Escande, A., Micaelli, A., Robert, T.: A hierarchical framework for realizing dynamically stable motions of humanoid robot in obstacle cluttered environments. In: IEEE International Conference on Humanoid Robots, pp. 867–874 (2012)
8. Hasunuma, H., Kobayashi, M., Moriyama, H., Itoko, T., Yanagihara, Y., Ueno, T., Ohya, K., Yokoil, K.: A tele-operated humanoid robot drives a lift truck. In: IEEE International Conference on Robotics and Automation, vol 3, pp. 2246–2252 (2002)
9. Hasunuma, H., Nakashima, K., Kobayashi, M., Mifune, F., Yanagihara, Y., Ueno, T., Ohya, K., Yokoil, K.: A tele-operated humanoid robot drives a backhoe. In: IEEE International Conference on Robotics and Automation, vol. 3, pp. 2998–3004 (2003)
10. Yokoi, K., Nakashima, K., Kobayashi, M., Mihune, H., Hasunuma, H., Yanagihara, Y., Ueno, T., Gokyuu, T., Endou, K.: A tele-operated humanoid operator. *Int. J. Robot. Res.* **25**(5–6), 593–602 (2006)
11. Pratt, G., Manzo, J.: The darpa robotics challenge - competitions. *IEEE Robot. Autom. Mag.* **20**(2), 10–12 (2013)
12. DARPA: Darpa robotics challenge trials 2013. Retrieved 2012, from <http://archive.darpa.mil/roboticschallengetrialsarchive>
13. DARPA: Darpa robotics challenge finals 2015. Retrieved 2015, from <http://www.theroboticschallenge.org>
14. Ackerman, E.: Darpa robotics challenge trials: What we learned on day 1. Retrieved December 21, 2013, from IEEE Spectrum
15. Hall, B.: Wpi robotics team finishes in seventh place in national competition. Retrieved December 22, 2013, from WPI News
16. Johnson, M., Shrewsbury, B., Bertrand, S., Wu, T., Duran, D., Floyd, M., Abeles, P., Stephen, D., Mertins, N., Lesman, A., Carff, J.: Team ihmco's lessons learned from the darpa robotics challenge trials. *J. Field Robot.* **32**(2), 192–208 (2015)
17. Paine, N., Mehling, J., Holley, J., Radford, N., Johnson, G., Fok, C., Sentis, L.: Actuator control for the nasa-jsc valkyrie humanoid robot: A decoupled dynamics approach for torque control of series elastic robots. *J. Field Robot.* **32**(3), 378–396 (2015)
18. DeDonato, M., Dimitrov, V., Du, R., Giovacchini, R., Knoedler, K., Long, X., Polido, F., Gennert, M., Padir, T., Feng, S., Moriguchi, H.: Human-in-the-loop control of a humanoid robot for disaster response: A report from the darpa robotics challenge trials. *J. Field Robot.* **32**(2), 275–292 (2015)
19. Kohlbrecher, S., Romay, A., Stumpf, A., Gupta, A., O, V.S., Bacim, F., Bowman, D., Goins, A., Balasubramanian, R., Conner, D.: Human-robot teaming for rescue missions: Team vigir's approach to the 2013 darpa robotics challenge trials. *J. Field Robot.* **32**(3), 352–377 (2015)
20. Yi, S., McGill, S., Vadakedathu, L., He, Q., Ha, I., Han, J., Song, H., Rouleau, M., Zhang, B., Hong, D., Yim, M.: Team thor's entry in the darpa robotics challenge trials 2013. *J. Field Robot.* **32**(3), 315–335 (2015)
21. Rasmussen, C., Sohn, K., Wang, Q., Oh, P.: Perception and control strategies for driving utility vehicles with a humanoid robot. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 973–980 (2014)
22. Rasmussen, C., Sohn, K., Yuvraj, K., Oh, P.: Early phases of humanoid vehicle ingress using depth cameras. In: IEEE International Conference on Technologies for Practical Robot Applications, pp. 1–6 (2014)
23. Sohn, K.: Optimization of vehicle mounting motions and its application to full-sized humanoid, DRC-Hubo. *J. Intell. Robot. Syst.*, 1–28 (2018)
24. Kim, S., Park., B., Park, J.: Drc finals 2015 - performance analysis of contestant teams. *Robot and Human (Korean Version)* **12**(4), 16–25 (2015)
25. Oh, P., Sohn, K., Jang, G., Jun, Y., Cho, b.: Technical overview of team drc-hubo@unlv's approach to the 2015 darpa robotics challenge finals. *J. Field Robot.* **34**(5), 874–896 (2017)
26. Lim, J., Lee, I., Shim, I., Jung, H., Joe, H.M., Bae, H., Sim, O., Oh, J., Jung, T., Shin, S., Joo, K.: Robot system of DRC-HUBO+ and control strategy of team KAIST in DARPA robotics challenge finals. *J. Field Robot.* **34**(4), 802–829 (2017)
27. Jung, T., Lim, J., Bae, H., Lee, K.K., Joe, H.M., Oh, J.H.: Development of the humanoid disaster response platform DRC-HUBO+. *IEEE Trans. Robot.* **34**(1), 1–17 (2018)
28. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **39**(1), 1–38 (1977)
29. Weinstein, A.J., Moore, K.L.: Pose estimation of ackerman steering vehicles for outdoors autonomous navigation. In: IEEE International Conference on Industrial Technology, pp. 579–584 (2010)
30. Cormen, T.H., Leiserson, C.E., Rivest, R.L.: Introduction to Algorithms. MIT Press and McGraw-Hill (1990)
31. PBS NOVA: Rise of the robots. Broad-casted on Feb 24th 2016, from <https://www.pbs.org/wgbh/nova/video/rise-of-the-robots>
32. Sohn, K., Markiewicz, M., Keilich, S.: Development of lower body for vehicle driving robot, hart. In: ASME International Mechanical Engineering Congress and Exposition. American Society of Mechanical Engineers Digital Collection (2019)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Kiwon Sohn received B.S. and M.S. degree of electrical engineering from Kyungpook National University (Daegu, South Korea) and University of Pennsylvania (PA, USA) in 2005 and 2007. He received Ph.D. degree of mechanical engineering from Drexel University (PA, USA) in 2014. He served as a Chief of Engineering in team DRC-HUBO@UNLV (the finalist of DARPA Robotics Challenge Finals) and DASL (Drones and Autonomous Systems Lab) in University of Nevada, Las Vegas, between 2014 and 2016. Currently, he is working in University of Hartford as an assistant professor. Prof. Sohn is a member of IEEE Robotics and Automation society.

Giho Jang received B.S. degree of computer science from Sunmoon University (Asan, South Korea) in 2007. He received M.S. and Ph.D. degree of electronic, electrical, control and instrumentation engineering from Hanyang University (Ansan, South Korea) in 2009 and 2015. He served as a lead soft engineer in team DRC-HUBO@UNLV (the finalist of DARPA Robotics Challenge Finals) and DASL (Drones and Autonomous Systems Lab) in University of Nevada, Las Vegas, between 2015 and 2016. Currently, he is working in Rainbow Robotics USA as a Chief of Research and Development. Dr. Jang is a member of IEEE Robotics and Automation society.