Toward Autonomous Driving by Musculoskeletal Humanoids

By Kento Kawaharazuka, Kei Tsuzuki, Yuya Koga, Yusuke Omura, Tasuku Makabe, Koki Shinjo, Moritaka Onitsuka, Yuya Nagamatsu, Yuki Asano, Kei Okada, Koji Kawasaki, and Masayuki Inaba

Digital Object Identifier 10.1109/MRA.2020.2987805
Date of current version: 12 May 2020

This article summarizes an autonomous driving project by musculoskeletal humanoids. The musculoskeletal humanoid, which mimics the human body in detail, has redundant sensors and a flexible body structure. These characteristics are suitable for motions with complex environmental contact, and the robot is expected to sit down on the car seat, step on the acceleration and brake pedals, and operate the steering wheel with both arms. We reconsider the developed

hardware and software of the musculoskeletal humanoid Musashi in the context of autonomous driving. The respective components of autonomous driving are conducted using the benefits of the hardware and software. Finally, we show that Musashi succeeded in the pedal and steering wheel operations with recognition.

Background

As a means of safe and comfortable transportation, various research studies in autonomous driving are in progress [1], [2]. Some companies have achieved a certain level of autonomous driving, e.g., headway vehicle following and autonomous parking. These cars are equipped with powerful cameras, lidar, GPS, and processors for accurate and safe control.

On the other hand, starting with the autonomous driving task at the DARPA Robotics Challenge (DRC) [3], there are research studies on autonomous driving by humanoid robots [4], [5]. Humanoid robots are equipped with various sensors for visual, acoustic, and force information. Using these sensors, the robot can get into the car and drive it. In addition, unlike ordinary autonomous driving, the humanoid robot is expected to do various other tasks, e.g., carrying heavy baggage, assisting the elderly, doing housework, and aiding with disaster response. However, because the humanoid robot lacks body flexibility and deviates from human body proportions, in the DRC, a steering wheel was operated by one arm and a special jig was required for sitting down on the seat.

Although various kinds of humanoid robots exist, one that imitates the human actuation system is the musculoskeletal humanoid [6]–[9]. The musculoskeletal humanoid is actuated not by motors arranged at each axis but by pneumatic actuators or muscle actuators with motors that imitate human muscles. The body is flexible compared with the ordinary axis-driven humanoid due to its muscle elasticity and underactuation, and thus it is suitable for motions with complex environmental contact. Therefore, the musculoskeletal humanoid can sit down on the car seat easily without a special jig and operate the steering wheel flexibly by both arms. Also, this robot can be used as a more realistic crash test dummy [10] because of its human-like body structure and actuation.

In this study, we introduce our project of autonomous driving by musculoskeletal humanoids (Figure 1). In particular, we describe the characteristics of the hardware and learning-based software to move the flexible body. We reconsider the developed hardware [9], [11]–[13] and software [14]–[16] in the context of autonomous driving. Also, we develop the respective components of autonomous driving and conduct experiments integrating them using the hardware and software characteristics.

Hardware of the Musculoskeletal Humanoid

Design Process

The important points for a robot to drive a car made for humans are considered to be 1) body proportion, 2) body flexibility, and 3) redundant sensors as well as a learning system using them. In this article, we first determine the arrangement of joints and the length of each bone to fit the proportion and joint structure of humans and attach muscle modules to the bones. In addition, a nonlinear elastic unit (NEU) is attached to the end of each muscle, and the robot can fit into the complex environment with the unit and the original flexibility of muscle wire. Finally, we design a structure including not only the muscle module that can measure muscle length, muscle tension, and muscle temperature but also a flexible hand with contact sensors, movable eyes with high-resolution cameras, and a foot with six-axis force sensors that can measure force along the entire surface as redundant sensors. In the section "Hardware Details," we describe in detail the design of muscle modules and NEUs that create body flexibility and the joint modules, eyes, hands, and feet that form the redundant sensors and learning system.

Hardware Details

We demonstrate the hardware overview of the developed musculoskeletal humanoid Musashi [9] in Figures 2 and 3. Currently, Musashi has 74 muscles and 39 joints, excluding the hand.

- Figure 2(a) shows the basic musculoskeletal structure of Musashi. The joint module connects the generic bone frames, and muscle modules are attached to the frame. The abrasion-resistant synthetic fiber Dyneema comes out as a muscle from the tension measurement unit of the muscle module and is folded back by the muscle relay unit. The NEU is attached to the end of the muscle, and a soft foam and spring cover the muscle. Muscles are antagonistically arranged around the joint. We describe muscles mainly conducting the current task as *agonist muscles* and the others as *antagonist muscles*.
- Figure 2(b) is the structure of the left arm of Musashi, constructed by the modules of 2(c)–(e) and explained subsequently. The body structure is easily constructed by the modules, muscle attachments connecting bones and muscles, and joint attachments connecting bones and joints.
- Figure 2(c) displays the detailed structure of the joint module. We can express all the basic human joints by rearranging two center parts and three axis parts. A

Figure 1. The musculoskeletal humanoids performing autonomous driving.

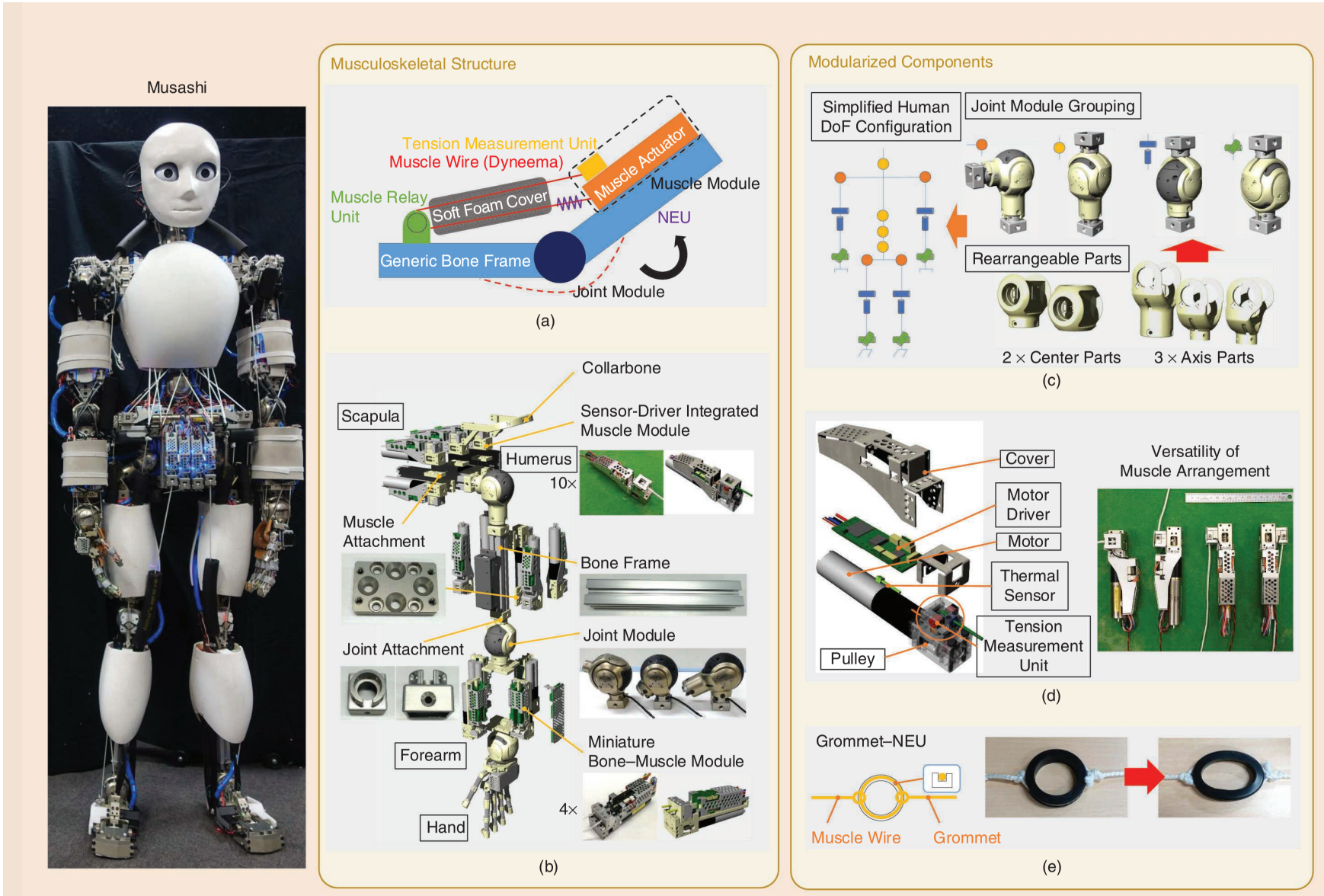


Figure 2. The overview of the developed musculoskeletal humanoid Musashi: the (a) basic musculoskeletal structure [9], (b) modularized structure of the left arm [9], (c) rearrangeable concept of joint modules [9], (d) detailed structure of muscle module (left figure) and versatility of muscle arrangement (right figure) [17], and (e) the NEU [9]. DoF: degrees of freedom.

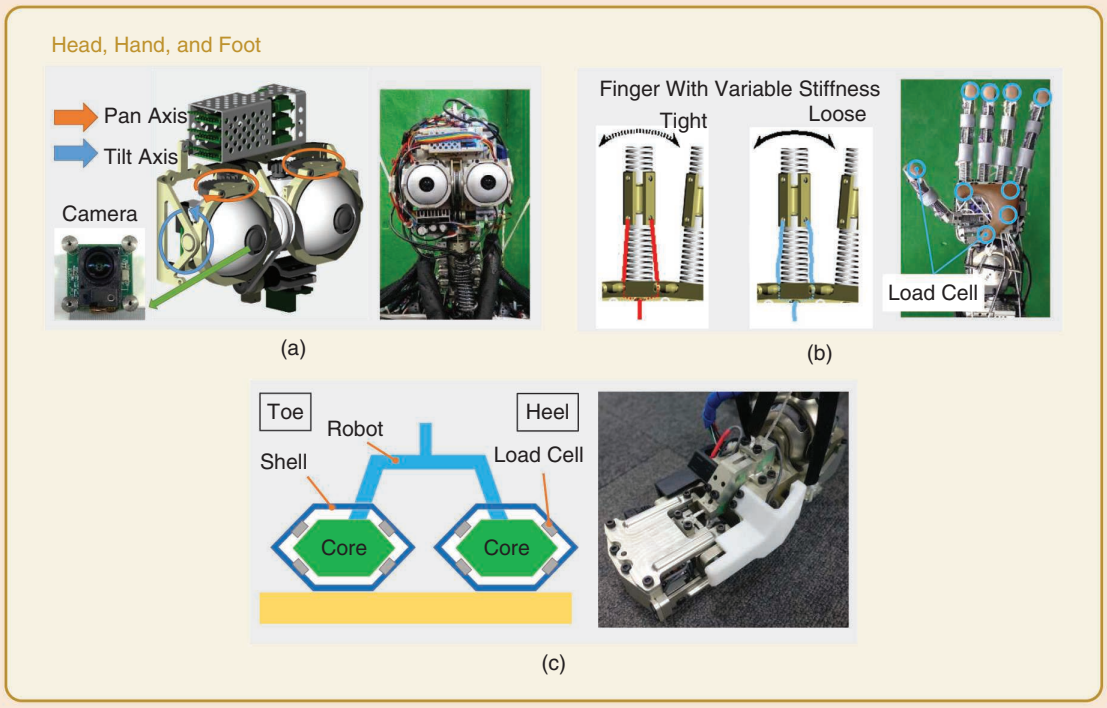


Figure 3. The overview of the developed musculoskeletal humanoid Musashi: (continued from Figure 2) The (a) movable eye unit [11], (b) flexible musculoskeletal hand [12], and (c) foot with six-axis core-shell structural force sensors (left: [13]).

potentiometer, inertial measurement units (IMUs), and a circuit integrating sensor data are packaged in the joint module. We can change the body structure easily by transforming this joint module.

- Figure 2(d) illustrates the detailed structure of the muscle module. We use two kinds of muscle modules depending on the body part. One is a sensor-driver integrated muscle module [17], which can drastically improve reliability and maintainability by packaging a motor, motor driver, thermal sensor, tension measurement unit, and so on into one module. The muscle is actuated by winding a muscle wire with a pulley. The other is a miniature bone-muscle module [18]. Although the basic concept is the same as in [17], the module is used not only as a muscle but also as a bone frame and dissipates heat to metal by packaging two smaller actuators into one module and filling the space between the two actuators with metal. We can get muscle tension, length, and temperature as sensor data. Also, as indicated in the right figure of 2(d), we can realize various muscle routes by rearranging the tension measurement unit.
- Figure 2(e) depicts the NEU. Previous NEUs are constructed by metal and springs, and the structure is not appropriate for environmental contact. Therefore, we realize the nonlinear elasticity using the compression of rubber, by covering the grommet rubber with Dyneema. Because the NEU is constructed with only rubber and Dyneema, the structure itself is flexible, and it is suitable for environmental contact.
- Figure 3(a) is the head of Musashi with the movable eye unit [11]. The unit includes three joints: a pan joint in each eye and a tilt joint. We use DFK-AFUJ003 (ImagingSource, Inc.) as the eyes, and we can change the image resolution, focus, exposure, and so forth. Although we considered using lidars/radars, we decided consistently to mimic the structure of human beings in detail.
- Figure 3(b) is the hand of Musashi with machined springs [12]. The hand does not break, even if hit by a hammer, because of the high flexibility. Muscles are antagonistically arranged at the proximal phalanges of each finger, and the finger stiffness can be changed by pulling the muscles and compressing the springs. Nine load cells are arranged at each fingertip and the palm of each hand, and contact force can be measured from them.
- Figure 3(c) is the foot of Musashi with six-axis core-shell force sensors [13]. The core-shell structure is arranged at the toe and heel, and multiple load cells are arranged between the core and shell. This structure can measure all of the force applied to the entire shell surface, and so the force to the instep of the foot can be measured.

Application of the Hardware to Autonomous Driving

We explain the use of the developed NEU, head, hand, and foot of Musashi in terms of autonomous driving in Figure 4 (presented in the multimedia material). Using NEUs, the

robot can not only make the body flexible but also change the stiffness freely. As demonstrated in Figure 4(a), the flexible body structure makes it possible to operate a steering wheel with both arms, which could not be seen in the DRC. Also, as depicted in Figure 4(b), the robot can change the impact response by changing its body stiffness. When dropping a 5-kg ball to the robot from a 1-m height, the maximum muscle tension is 150 N with low stiffness and 250 N with high stiffness. This resembles the increase of human arm stiffness during a car crash, and the use of Musashi as a crash test dummy is anticipated.

Using the head with the movable eye unit, the robot can recognize objects with a wide field of view as indicated in Figure 4(c). Also, the robot can recognize a human in a side mirror using the high-resolution camera, as shown in Figure 4(d). Using the flexible hand with machined springs, the robot can adapt the grasp shape to various situations inside the car, such as pulling a handbrake [Figure 4(e)] and rotating a key [Figure 4(f)]. Also, using the variable-stiffness mechanism of fingers, as displayed in Figure 4(g), the robot can operate a blinker lever correctly with high stiffness. Using the foot, which can measure the force applied to its entire surface, the robot can

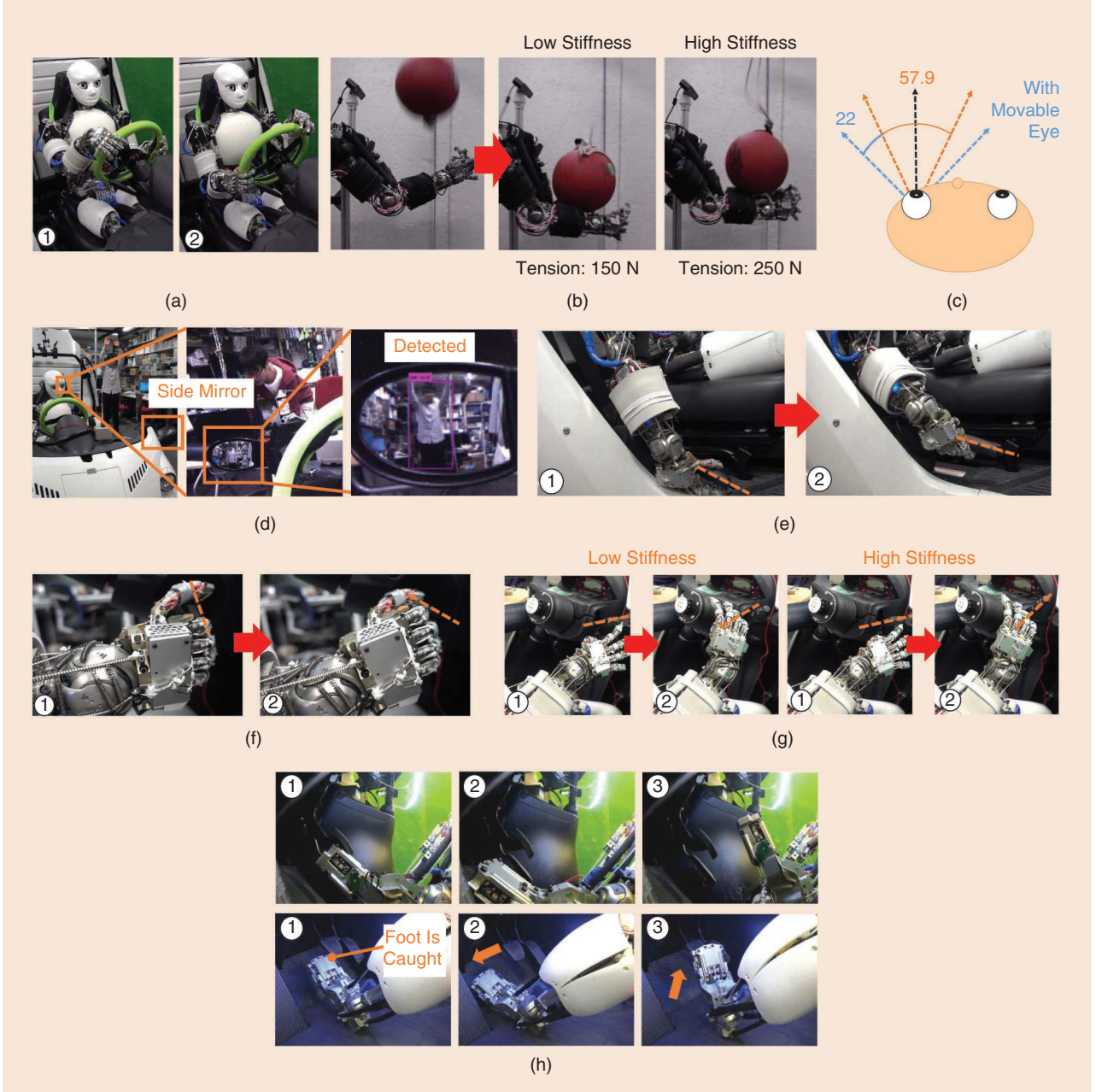


Figure 4. The realization of the respective components of autonomous driving using the characteristics of the developed hardware: (a) the steering wheel operation with both arms, (b) the variable-stiffness control using the NEUs, (c) the field of view of the movable eye unit, (d) a human detection experiment in the side mirror, (e) pulling a handbrake, (f) rotating a key, (g) operating a blinker lever by changing the stiffness of fingers, and (h) recovering from slipping during brake pedal operation using the developed foot.

recover the foot position by sensing the force to its instep when slipping during a brake pedal operation, as presented in Figure 4(h).

Software of the Musculoskeletal Humanoid

Design Process

To handle the flexible body and redundant sensors, we consider that a learning-based motion generation, learning-based recognition, and fast reflex control are necessary. Learning-based motion generation can be divided into two types of motion generators, for static behavior and dynamic behavior. In the former, the dimension of the state space is relatively small, and online learning is effective. In the latter, the dimension is relatively large, and only offline learning can be used currently. In addition, we also develop learning-based recognition methods using the eye and ear sensors because visual and sound recognition is significant in performing the driving task. Since it is difficult to operate learning-based controls at high frequency, a safety mechanism should be implemented in a lower layer as reflex control. In this study, we use a simple conditional branching based on the recognition results for motion planning.

Software Details

We show the overview of the learning-based software system in Figure 5. We describe the current muscle length as $\boldsymbol{l}$, current muscle tension as $\boldsymbol{f}$, current joint angle as $\boldsymbol{\theta}$, current muscle temperature as $\boldsymbol{c}$, the muscle Jacobian as $\boldsymbol{G}$, current image as $\boldsymbol{I}$, current sound as $\boldsymbol{S}$, task state as $\boldsymbol{s}_{\text{task}}$, current state obtained from vision as $\boldsymbol{s}_{\text{vision}}$, and current state obtained from sound as $\boldsymbol{s}_{\text{sound}}$. Also, $\{\boldsymbol{l}, \boldsymbol{\theta}, \boldsymbol{f}, \boldsymbol{s}_{\text{task}}\}^{\text{ref}}$ is the control command of each state, and $\boldsymbol{\theta}^{\text{est}}$ is the estimated joint angle.

The basic components of this system are the intersensory network module (static module) [Figure 5(a)], dynamic task control network module (dynamic module) [Figure 5(b)], reflex module [Figure 5(c)], and recognition module [Figure 5(d)]. The static module [Figure 5(a)] acquires the following function $\boldsymbol{h}_{\text{static}}$ [14]:

$$\boldsymbol{l} = \boldsymbol{h}_{\text{static}}(\boldsymbol{\theta}, \boldsymbol{f}). \tag{1}$$

This represents the static relationship among $\boldsymbol{\theta}, \boldsymbol{l}$, and $\boldsymbol{f}$. When $\boldsymbol{\theta}^{\text{ref}}$ and $\boldsymbol{f}^{\text{ref}}$ are given depending on the task, by inputting them

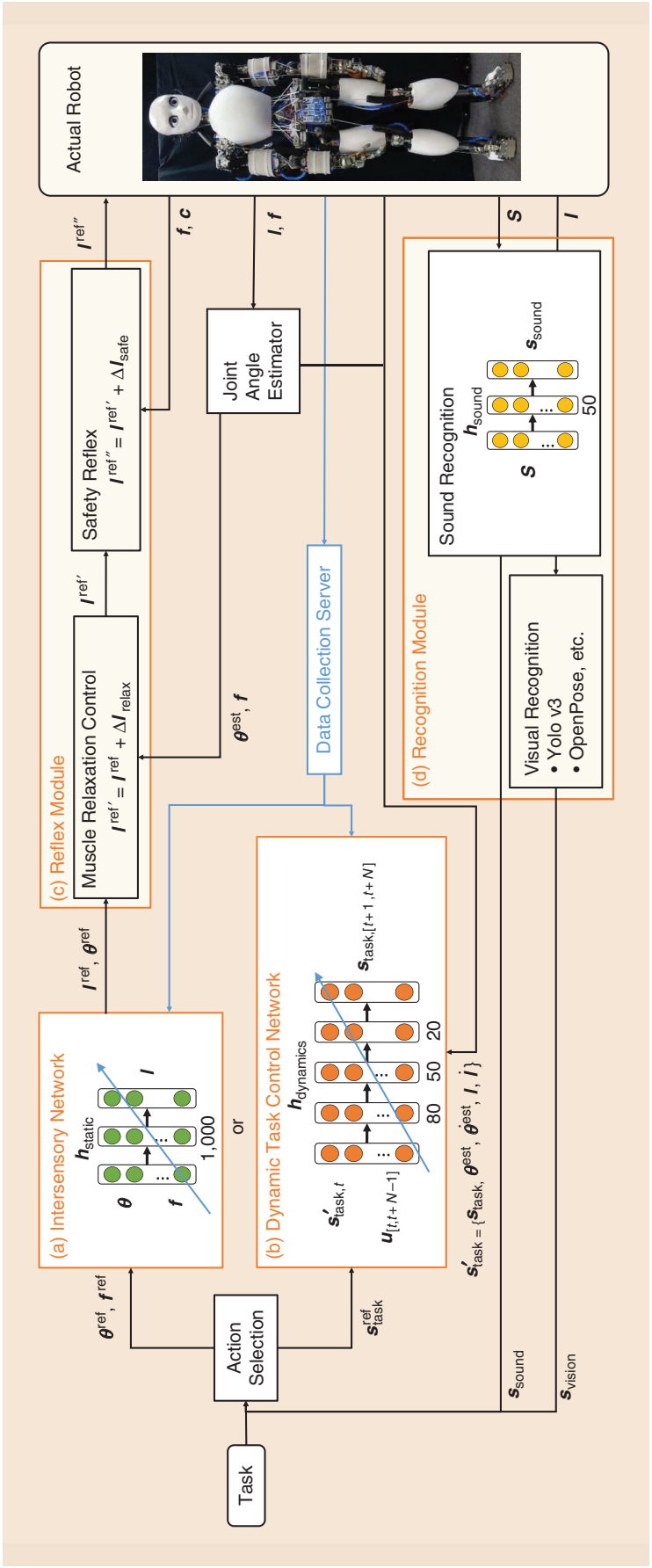


Figure 5. The overview of the developed software with four modules: the (a) intersensory network module (static module), (b) dynamic task control network module (dynamic module), (c) reflex module, and (d) recognition module.

into (1), $\boldsymbol{l}^{\text{ref}}$ to send to the robot is calculated. This function is expressed by a neural network and is initialized by human-made data. Then, we update the network online by using the actual robot sensor information $(\boldsymbol{\theta}, \boldsymbol{f}, \boldsymbol{l})$. At every movement, the network is updated, and the robot becomes able to realize $\boldsymbol{\theta}^{\text{ref}}$, $\boldsymbol{f}^{\text{ref}}$ accurately. In this study, $\boldsymbol{\theta}$ can be obtained from the joint module, but the ordinary musculoskeletal humanoid does not have joint angle sensors due to complex joints, such as the ball or scapula joints. In that case, $\boldsymbol{\theta}$ must be obtained from a motion capture or vision sensor. Also, by using this network $\boldsymbol{h}_{\text{static}}$, not only control but also estimation of joint angles $\boldsymbol{\theta}^{\text{est}}$ are enabled by using an extended Kalman filter and the change in muscle length and tension.

The dynamic module [Figure 5(b)] acquires the function $\boldsymbol{h}_{\text{dynamic}}$ [15]

$$\boldsymbol{s}_{\text{task},[t+1,t+N]} = \boldsymbol{h}_{\text{dynamics}}(\boldsymbol{s}'_{\text{task},t}, \boldsymbol{u}_{[t,t+N-1]}), \tag{2}$$

where $\{\boldsymbol{s}_{\text{task}}, \boldsymbol{u}\}_{[t_1,t_2]}$ is a vector vertically arranging $\{\boldsymbol{s}_{\text{task}}, \boldsymbol{u}\}$ from t_1 to t_2 time steps, $\boldsymbol{s}'_{\text{task},t}$ is the initial task state ($\{\boldsymbol{s}_{\text{task}}, \boldsymbol{\theta}^{\text{est}}, \dot{\boldsymbol{\theta}}^{\text{est}}, \boldsymbol{l}, \dot{\boldsymbol{l}}\}$ in this study), $\boldsymbol{u}$ is the control command ($\boldsymbol{\theta}^{\text{ref}}$ or $\boldsymbol{l}^{\text{ref}}$ in this study), and N is the number of time steps to expand this state equation. This function is equivalent to a network representing a dynamic transition of the task state by a control command sequence. This network can be trained by gathering data of the observed task state transition when sending random control commands. When realizing a certain task state $\boldsymbol{s}^{\text{ref}}_{\text{task}}$, we execute the following equations after setting the initial $\boldsymbol{u}$:

$$\boldsymbol{s}^{\text{pred}}_{\text{task,seq}} = \boldsymbol{h}_{\text{dynamic}}(\boldsymbol{s}'_{\text{task},t}, \boldsymbol{u}^{\text{init}}_{\text{seq}}), \tag{3}$$

$$L = \text{MSE}(\boldsymbol{s}^{\text{pred}}_{\text{task,seq}}, \boldsymbol{s}^{\text{ref}}_{\text{task,seq}}) + \alpha E_{\text{adj}}(\boldsymbol{u}^{\text{init}}_{\text{seq}}), \tag{4}$$

$$\boldsymbol{g} = dL/d\boldsymbol{u}^{\text{init}}_{\text{seq}}, \tag{5}$$

$$\boldsymbol{u}^{\text{init}}_{\text{seq}} \leftarrow \boldsymbol{u}^{\text{init}}_{\text{seq}} - \beta\frac{\boldsymbol{g}}{|\boldsymbol{g}|}, \tag{6}$$

where $\boldsymbol{u}^{\text{init}}_{\text{seq}}$ is the sequence of the initial value of control command, $\boldsymbol{s}^{\text{pred}}_{\text{task,seq}}$ is the predicted sequence of $\boldsymbol{s}_{\text{task}}$ in $[t+1, t+N]$ calculated from $\boldsymbol{s}'_{\text{task,t}}$ and $\boldsymbol{u}^{\text{init}}_{\text{seq}}$, and $\boldsymbol{s}^{\text{ref}}_{\text{task,seq}}$ is a vector arranging N number of $\boldsymbol{s}^{\text{ref}}_{\text{task}}$. Also, MSE is mean squared error, E_{adj} is the MSE of the values at adjacent time steps, α is a weight constant, and β is an update rate. We calculate the loss between the predicted and target $\boldsymbol{s}_{\text{task}}$, add the loss to smooth the target control command sequence, and update the control command sequence by backpropagation [19]. By repeating this procedure, $\boldsymbol{u}^{\text{init}}_{\text{seq}}$ is updated to accurately realize $\boldsymbol{s}^{\text{ref}}_{\text{task}}$, and the task is executed by sending $\boldsymbol{u}^{\text{init}}_{t}$ to the actual robot. While Figure 5(a) shows the network regarding static motions, Figure 5(b) is the network regarding dynamic motions, and its learning is difficult due to more variables. Thus the robot is moved mainly by the mechanism of Figure 5(a), but the mechanism in Figure 5(b) is constructed offline and used when conducting tasks in which dynamic and accurate motions are required.

The reflex module [Figure 5(c)] is executed at high frequency. Muscle relaxation control (MRC) [16] is a control that elongates muscles from antagonist muscles to agonist muscles while keeping the current joint angle. First, we calculate the necessary muscle tension $\boldsymbol{x}$ that achieves the necessary joint torque $\boldsymbol{\tau}^{\text{nec}}$ by solving quadratic programming:

$$\underset{\boldsymbol{x}}{\text{minimize}} \quad \boldsymbol{x}^T W_1 \boldsymbol{x} + (G^T\boldsymbol{x} + \boldsymbol{\tau}^{\text{nec}})^T W_2 (G^T\boldsymbol{x} + \boldsymbol{\tau}^{\text{nec}}), \tag{7}$$

$$\text{subject to} \quad \boldsymbol{x} \geq \boldsymbol{f}^{\min}, \tag{8}$$

where W_1, W_2 are weight matrices and $\boldsymbol{f}^{\min}$ is the minimum muscle tension. We sort muscles by $\boldsymbol{x}$ in ascending order and gradually elongate them in order starting with unnecessary antagonist muscles with smaller tension. In detail, we gradually increase the muscle relaxation value $\Delta\boldsymbol{l}_{\text{relax}}$ and add it to the muscle length to send to the actual robot. When the muscle tension becomes smaller than $\boldsymbol{f}^{\min}$, the muscle to elongate is changed to the next muscle, and this control stops when the current joint angle is changed more than a threshold. This procedure works in a static state. At a moving state, we sort muscles by $\boldsymbol{x}$ in descending order and gradually decrease $\Delta\boldsymbol{l}_{\text{relax}}$ starting with necessary muscles. This reflex can inhibit unnecessary muscle tension of antagonist muscles due to the model error. Also, for example, when resting the arms on the table, the body and environment are constrained, and so not only the antagonist muscle tension but also the agonist muscle tension can be reduced because the joint angle does not change even when relaxing agonist muscles. Thus, the robot becomes able to rest its body and move continuously for a longer time. Safety reflex is a simple control that elongates muscle length in order not to break motors due to high muscle tension and temperature. We add $\Delta\boldsymbol{l}_{\text{safe}}$ calculated to the target muscle length,

$$\Delta\boldsymbol{l}^{\text{ref}}_{\text{safe}} = K_f \max(\boldsymbol{f} - \boldsymbol{f}^{\text{lim}}, \boldsymbol{0}) + K_c \max(\boldsymbol{c} - \boldsymbol{c}^{\text{lim}}, \boldsymbol{0}), \tag{9}$$

$$\Delta\boldsymbol{l}_{\text{safe}} \leftarrow \Delta\boldsymbol{l}_{\text{safe}} + \max(\Delta\boldsymbol{l}^{\min}, \min(\Delta\boldsymbol{l}^{\max}, \Delta\boldsymbol{l}^{\text{ref}}_{\text{safe}} - \Delta\boldsymbol{l}_{\text{safe}})), \tag{10}$$

where $\{\boldsymbol{f}, \boldsymbol{c}\}^{\text{lim}}$ is a threshold of muscle tension or temperature to begin the reflex, $K_{\{f,c\}}$ is a gain, $\Delta\boldsymbol{l}^{\text{ref}}_{\text{safe}}$ is an ideal elongation value, and $\Delta\boldsymbol{l}^{\{\min,\max\}}$ is a minimum or maximum change of $\Delta\boldsymbol{l}_{\text{safe}}$ at one time step. By elongating the muscle length considering muscle tension and temperature, we can inhibit the burnout of motors.

The recognition module [Figure 5(d)] is divided into object and sound recognition. In this study, we use Yolo v3 [20] for object recognition. The labels mainly used in this study are "car," "person," and "traffic light." Also, regarding sound recognition, the sound is converted into a mel spectrum, and the network $\boldsymbol{h}_{\text{sound}}$ is trained to output sound class from the spectrum. In this study, this module recognizes whether or not the sound is a car horn.

Application of the Software to Autonomous Driving

We present the use of each software component for autonomous driving in Figure 6 (presented in the multimedia

material). The static module can be applied to various motions, and is especially useful for steering wheel operation. In Figure 6(a), we depict an experiment for operating the steering wheel during online learning of the static module. The graphs of Figure 6(a) illustrate the transition of the steering wheel angle and 10 muscle tensions of the shoulder and elbow of Musashi. At every operation, the angle gradually increases, and muscle tension gradually decreases. Thus, the relationship among $\boldsymbol{f}$, $\boldsymbol{\theta}$, and $\boldsymbol{l}$ is correctly updated using the actual robot sensor information.

The dynamic module is useful for dynamic motions. The pedal operation is a good example because a fast motion adaptation is necessary. In Figure 6(b), we trained the dynamic module using the data of random pedal operation for one minute and conducted pedal operation with the trained network. This experiment was conducted indoors, and the rear wheels were on free rollers for safety. The car velocity was obtained through the controller area network-USB of the car. We describe the car velocity as v_{car} and the joint angle of the right ankle pitch as $\boldsymbol{\theta}_{\text{ankle}}$, and we set $\boldsymbol{\theta}_{\text{ankle}}$ as $\boldsymbol{u}$. We set $v_{\text{car}}^{\text{ref}} = 5\,[\text{km/h}]$ and conducted the manually tuned proportional-integral-derivative (PID) control and the proposed control with the dynamic module (proposed). When comparing PID and proposed, the error between v_{car} and $v_{\text{car}}^{\text{ref}}$ falls within 20% starting from 22.9 s and 0.9 s, respectively. Since the dynamic relationship between the joint angle of ankle pitch and car velocity is complex, this was the limit of PID by manual tuning, and the car velocity vibrated significantly when the gain in our experiment was further increased. Thus, by acquiring the state equation between task state and control command, fast tracking is enabled.

The reflex module is useful for motions with long rest time. For example, as shown in Figure 6(c), during the steering wheel operation, the robot does not always operate the wheel. In this case, MRC gradually makes antagonist muscles elongate and internal force

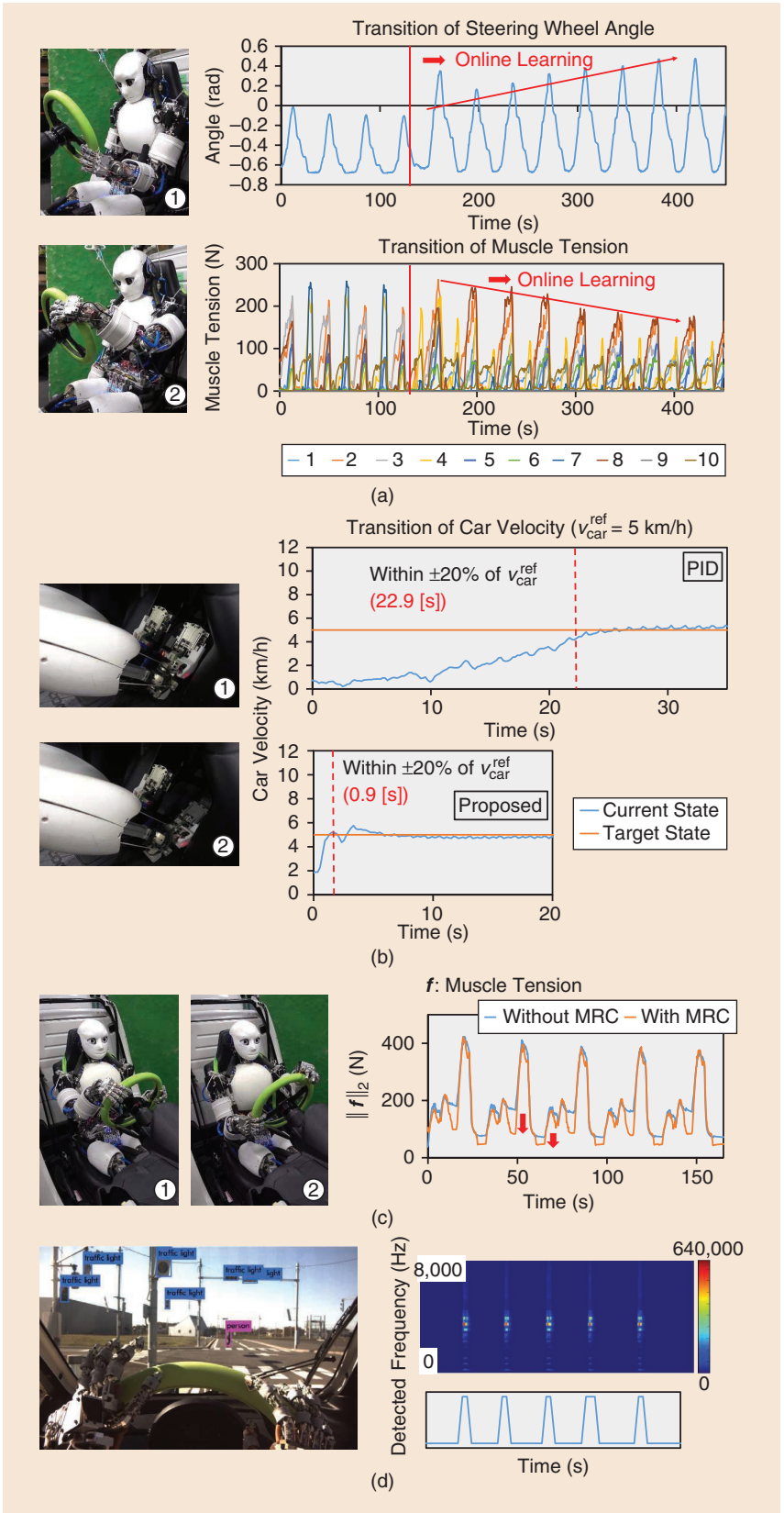


Figure 6. The respective components of autonomous driving using the developed software: (a) the steering wheel operation experiment using the online learning of static module [14], (b) the pedal operation experiment using the trained dynamic module [15], (c) the steering wheel operation experiment with and without MRC [16], and (d) visual recognition of traffic lights and a human and sound recognition of a car horn.

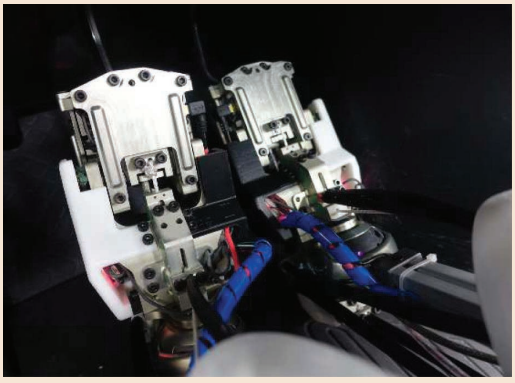
decrease. Also, because the steering wheel and the hands are constrained, the current joint angle does not change significantly even when agonist muscles are elongated. We demonstrate the transition of L2 norm of 10 muscle tensions of the elbow and shoulder in the left arm of Musashi $\|\boldsymbol{f}\|_2$, with and without MRC. The muscle tension is reduced in a static state with MRC compared to without MRC. This module can inhibit the increase of muscle temperature, and long-time operation is enabled.

The recognition module is mainly used as visual recognition of "person" and "traffic light," and sound recognition of the car horn. The left figure of Figure 6(d) depicts the recognition result at a crossing, and the traffic lights and a human are recognized well. Also, the right-hand side of Figure 6(d) displays the sound spectrum of a car horn, and the car horn is recognized well.

Experiments

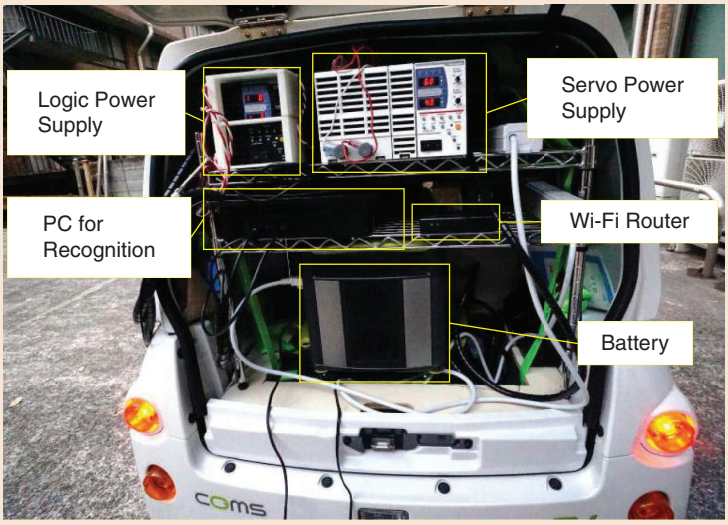
Experimental Setup

The car used in this study is a B.COM Delivery of the extremely small electric vehicle COMS (Chotto Odekake Machimade Suisui) series. For safety, its motor torque is limited to 5 Nm, and an emergency stop button is equipped. In the pedal operation, as indicated in Figure 7(a), the acceleration and brake pedals are operated by the right and left foot, respectively. The experiment was conducted on the Kashiwa Campus at the University of Tokyo, as presented in Figure 7(b). As shown in Figure 7(c), the COMS is equipped with a battery, logic power supply, servo power supply, Wi-Fi router, and PC for the recognition module in the trunk. The COMS is an electric vehicle, and all of the electric power for the robot can be obtained from COMS models in the future.

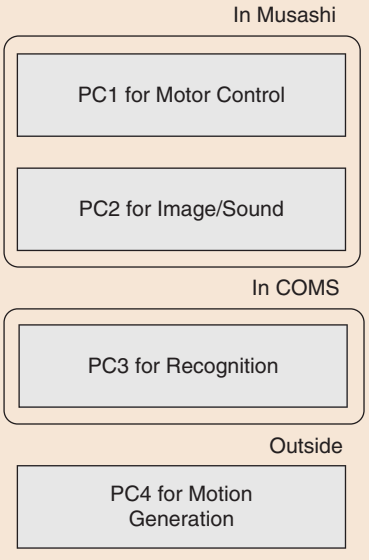


(a)

(b)



(c)



(d)

Figure 7. The experimental setup of autonomous driving by the musculoskeletal humanoid Musashi: the (a) pedal operation configuration, (b) experimental environment, (c) experimental configuration in the COMS [15], and (d) configuration of the PCs.

As depicted in Figure 7(d), PCs (Intel NUC, Intel, Inc.) for motor control and image/sound are in the head of Musashi; the recognition is executed on the PC (ZOTAC VR GO, ZOTAC, Inc.) in the COMS, and the other processes are executed on PC4 outside.

Pedal Operation With Recognition

We conducted an experiment integrating the pedal operation and recognition (Figure 8; presented in the multimedia material). We display the experimental motion flow in Figure 8(a). First, the robot operates the acceleration pedal using the trained dynamic module, steps on the brake pedal if it recognizes a human, restarts the acceleration pedal operation, and steps on the brake pedal again if it recognizes a car horn. The motion sequence is depicted in Figure 8(b), and we can see that it succeeded. The transition of car velocity (upper graph) and joint angle of the right ankle pitch (lower graph) is presented in Figure 8(c), while Figure 8(d) demonstrates the recognition result of a human. The robot recognizes a human when its bounding box is larger than a certain threshold and its center coordinate is around the center of the image. Figure 8(e) provides the recognition result of a car horn. When a human [Figure 8(c)] or car horn [Figure 8(d)] was detected, the brake pedal was stepped on, and we can see that the car velocity became 0. The big problem here is the poor tracking of the target car velocity as illustrated in Figure 8(c). This is because of the difference between the environment that the network is trained in and the experimental environment. When $t = [0,60]$ [sec], the car velocity decreased because the road friction was high, and when $t = [90,110]$ [sec], the car velocity increased because the road was downhill. This was not a problem in the experiment on the free roller, but the difference of the actual environment was difficult to handle. Because the learning-based controls depend on the data used for training, we must continue to conduct experiments outside, obtain data, and solve this problem.

Handle Operation With Recognition

We conducted the steering wheel operation with recognition (Figure 9; presented in multimedia material). We demonstrate the motion flow in Figure 9(a). The robot steps on the brake pedal when the traffic light is red, releases the brake pedal when the traffic light turns blue, and turns right at the crossing by the steering wheel operation with the static module. In this experiment, because turning at a curve is difficult when the car velocity is fast, we use creep velocity just by releasing the brake pedal. Also, the steering wheel is turned to the right first and then is turned to the left by a human command.

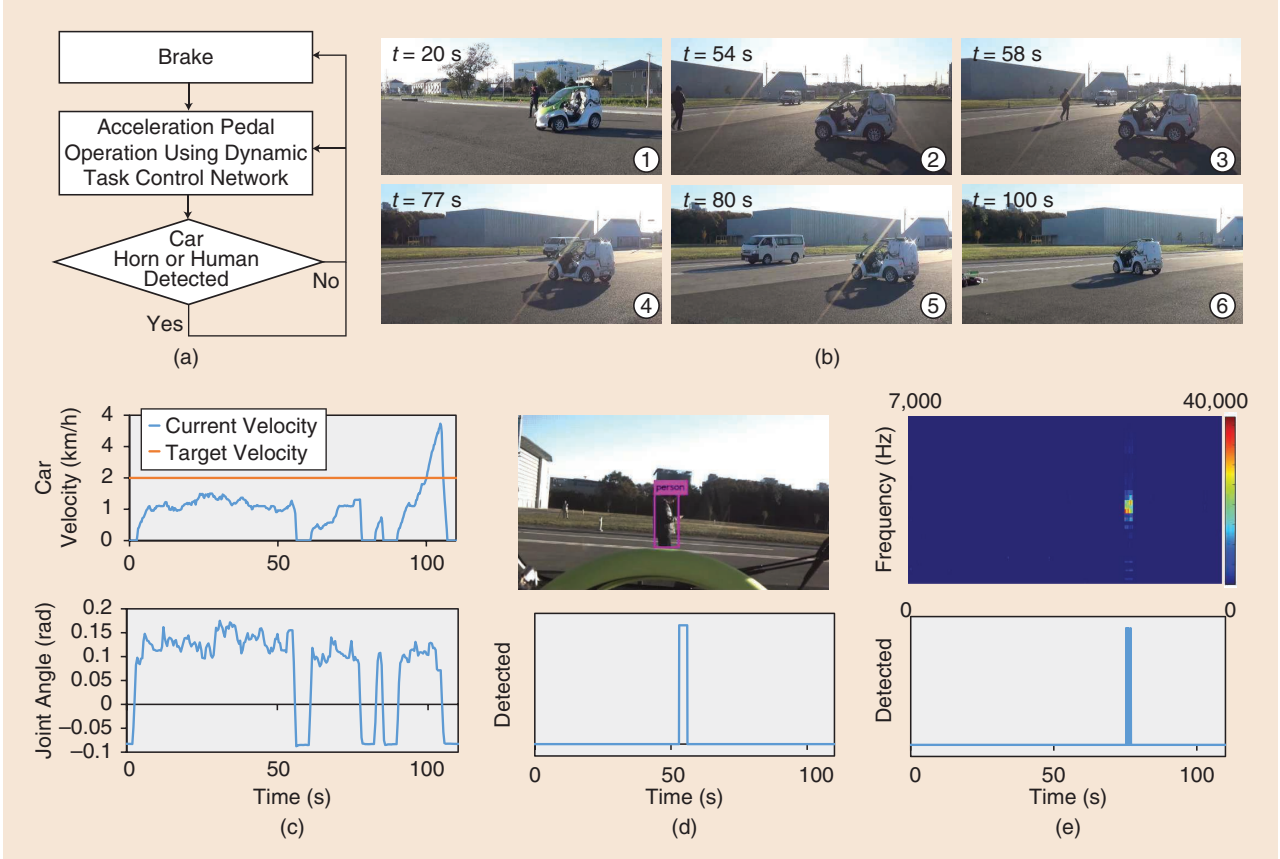


Figure 8. The pedal operation experiment with recognition: the (a) experimental motion flow, (b) experimental appearance, (c) current and target car velocity (upper graph) and joint angle of right ankle pitch for the acceleration pedal (lower graph), (d) visual recognition result, and (e) sound spectrum and its recognition result.

Figure 9(b) depicts the overview of this experiment, and we can see that the car began to move when the traffic light turned blue and the car could turn right at the crossing by 90°. The sequence of the steering wheel operation is displayed in Figure 9(c). The sequence is rotating the steering wheel by both arms as much as possible, releasing the left hand, returning the left hand to the original position, releasing the right hand, returning the right hand to the original position, and rotating again. The transition of the steering angle is presented in Figure 9(d). The robot could turn the steering wheel by about 180° in 70 s. Figure 9(e) shows the recognition result of the traffic light. The light is cropped by the object detection module, and the blue or red of the traffic light is recognized based on the ratio of red and blue pixels. The module could recognize the moment when the light turns from blue to red. At the same time, as shown in Figure 9(f), the left ankle pitch joint moved to release the brake pedal, and the car began to move. The problem of this experiment is the slowness of the steering wheel operation. Currently, turning at the crossing takes about two minutes, and we must make the motions faster and smoother.

By making use of flexibility, variable-stiffness structure, and several sensors, we succeeded in the steering wheel operation with both arms and human recognition in the side mirror.

Limitations and Future Works

Pedal Operation

The pedal operation is one of the tasks with various remaining issues. In this study, we assumed a flat and smooth road and developed a method to achieve the target car velocity quickly by representing the state equation between the car velocity and joint angle of the ankle pitch. However, there are three issues.

First, in the actual driving environment, the road is not smooth; the ground rises at a crossing, and the road is sometimes uphill or downhill. In those cases, the state equation trained at a flat road is different from that of the actual environment, and the robot cannot adjust the car velocity well. To solve this problem, we need to conduct online learning or add the image of road condition and IMU information in the body to the initial task state $\boldsymbol{s}^{\text{task}'}$. However, online learning becomes difficult with additional network input, and an efficient learning system with only a few data is desired.

Second, the robot adjusts only the acceleration pedal and cannot adjust the brake pedal. When driving more slowly than creep velocity, the robot must adjust the car velocity by stepping on the brake pedal. Also, in this study, although not required at the slow car velocity, the robot needs to acquire how to smoothly step on the brake pedal as the car velocity becomes fast.

Third, the car velocity is currently obtained from the COMS software, but it should be obtained using image information, IMUs, and so forth. Since the car velocity obtained from visual odometry is too noisy for pedal

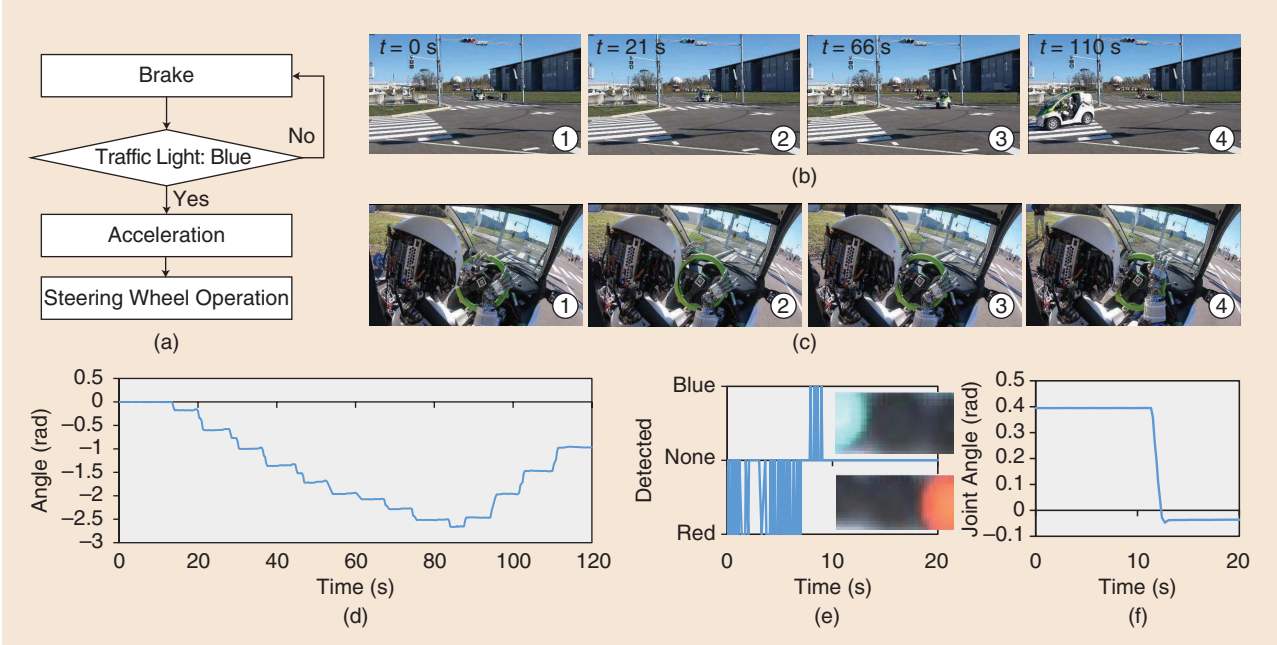


Figure 9. The steering wheel operation with recognition: the (a) experimental motion flow, (b) experimental appearance, (c) sequence of steering wheel operation, (d) transition of the steering wheel angle, (e) traffic light recognition result, and (f) joint angle of the left ankle pitch for brake pedal operation.

operation now, especially at low speed, we need to develop a new estimation algorithm.

Steering Wheel Operation

Although Musashi succeeded in the steering wheel operation with both arms for the first time, there remain some issues. First, in this study, the robot operated the steering wheel not by grasping it but by pressing the hand to it. Releasing and holding the steering wheel were difficult because the fingers sometimes were caught by the wheel or the robot could not accurately grasp the intended part. The first problem is prominent because the complex finger structure is often caught somewhere in the car, and the robot cannot restore itself. The hand structure with smooth skin or a glove may be able to solve this problem. Also, the robot needs to identify from sensors the direction its hand was caught and restore itself.

Second, in this study, the robot operated the steering wheel by a push–pull steering method, and it is not the usual human motion of the cross-arm steering. In the cross-arm steering, the arms move while interfering with each other, and the inverse kinematics must be solved in a wide range. We experienced that large internal force among the arms sometimes emerges and parts of the hands were caught by each other. Also, we found that the robot must have the scapula joint to solve the inverse kinematics in a wider range.

Recognition

We used the general object detection method for the visual recognition module, and the recognition of the road line and distance to objects are not conducted. Particularly, the latter is due to the structure of the developed eye unit. While the movable eye unit can enlarge the field of view by moving the eyes, the external parameter changes by this movement, and estimation of the depth map by stereo vision is difficult. Currently, we are developing a learning-based method to estimate the object distance from the convergence angle, binocular parallax, and so on. Also, our experiments are conducted only at daytime. At nighttime, although we humans can recognize traffic lights and pedestrians through the cameras, Musashi (that is, Yolo v3) cannot. We used the pretrained model of Yolo v3, but the model does not use time-series transition of image and is not trained using the dataset at the night time. Therefore, we need to train another model using video and dataset at night.

Although acoustic information is rarely used for autonomous driving, we applied it to the recognition of the car horn. In the future, we need to develop a system for the robot to understand the current situation by integrating the visual and sound information, recognize the anomaly of the engine, and talk with humans.

The recognition of the failure of the hardware components will also become important. In musculoskeletal humanoids, the failure is mainly in the muscle, which can be broken when strong force is applied continuously or when the wire is frayed by friction. By using an autoencoder-type network, we can detect anomalies; even if the muscle is ruptured, by continuing online learning, we can obtain the intersensory network in the broken state as in [14]. However, the actual task of autonomous driving requires quicker adaptation and operation in dangerous situations, and the detection of anomalies in hardware components and their handling should be considered more deeply for safety.

Future Works

The content we handled in this study is only a part of the entire system for autonomous driving. The robot must get into the car, localize the self-position with simultaneous localization and mapping, and conduct a driving motion plan. Also, the respective components of rotating a key, pulling a handbrake, looking around, operating an acceleration pedal, and operating a steering wheel must be integrated into one system. For this purpose, we need a method that handles the flexible body, manages the muscle temperature, and recognizes the situation more accurately. Also, we need to improve the hardware for the scapula with wide range motion and the body shape with smooth human-like skin.

Furthermore, to be able to drive not only a single car but also various cars, we have to consider what needs to be learned immediately once a car is changed. It is necessary to learn not only the current intersensory network and dynamic task control network but also the position of the key, side brake, and so forth in the future.

Finally, this is the first example of a robot sitting in a car and driving it with both arms without a jig, but some metric is needed to compare its performance in the future. As for performance, we can think of an index of how long it takes to complete a certain course. However, there are many difficult problems, such as what kind of course to make and how neatly the course should be driven. As for safety, it is more difficult to establish an index. Although a long-term endurance test is possible, we have not yet reached the stage.

Conclusion

In this study, we focused on autonomous driving by the musculoskeletal humanoid Musashi using the characteristics of its hardware and software. By making use of flexibility, variable-stiffness structure, and several sensors, we succeeded in the steering wheel operation with both arms and human recognition in the side mirror. Also, we proposed a learning-based system handling the flexible body with difficult modeling and succeeded in the pedal and steering wheel operations with recognition. For autonomous driving by humanoids in the future, we would like to develop the next hardware and software using the obtained knowledge.

References

[1] J. Levinson et al., "Towards fully autonomous driving: Systems and algorithms," in *Proc. 2011 IEEE Intelligent Vehicles Symp. (IV)*, pp. 163–168. doi: 10.1109/IVS.2011.5940562.

[2] M. R. Endsley, "Autonomous driving systems: A preliminary naturalistic study of the tesla model S," *J. Cognit. Eng. Decis. Making*, vol. 11, no. 3, pp. 225–238, 2017. doi: 10.1177/1555343417695197.

[3] "DARPA Robotics Challenge," DARPA, Arlington, VA. Accessed on: Mar. 16, 2020. [Online]. Available: https://archive.darpa.mil/roboticschallenge/
[4] C. Rasmussen, K. Sohn, Q. Wang, and P. Oh, "Perception and control strategies for driving utility vehicles with a humanoid robot," in *Proc. 2014 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2014, pp. 973–980. doi: 10.1109/IROS.2014.6942677.
[5] A. Paolillo, P. Gergondet, A. Cherubini, M. Vendittelli, and A. Kheddar, "Autonomous car driving by a humanoid robot," *J. Field Robot.*, vol. 35, no. 2, pp. 169–186, 2018. doi: 10.1002/rob.21731.
[6] H. G. Marques et al., "ECCE1: The first of a series of anthropomimetic musculoskeletal upper torsos," in *Proc. 2010 IEEE-RAS Int. Conf. Humanoid Robots*, pp. 391–396. doi: 10.1109/ICHR.2010.5686344.
[7] Y. Nakanishi et al., "Design approach of biologically-inspired musculoskeletal humanoids," *Int. J. Adv. Robot. Syst.*, vol. 10, no. 4, pp. 216–228, 2013. doi: 10.5772/55443.
[8] Y. Asano et al., "Human mimetic musculoskeletal humanoid Kengoro toward real world physically interactive actions," in *Proc. 2016 IEEE-RAS Int. Conf. Humanoid Robots*, pp. 876–883. doi: 10.1299/jsmermd.2016.2A1-13a2.
[9] K. Kawaharazuka et al., "Component modularized design of musculoskeletal humanoid platform Musashi to investigate learning control systems," in *Proc. 2019 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2019, pp. 7294–7301. doi: 10.1109/IROS40897.2019.8968068.
[10] E. Haug, H. Choi, S. Robin, and M. Beaugonin, "Human models for crash and impact simulation," in *Computational Models for the Human Body* (Handbook of Numerical Analysis Series). New York: Elsevier, 2004, vol. 12, pp. 231–452.
[11] T. Makabe et al., "Development of movable binocular high-resolution eye-camera unit for humanoid and the evaluation of looking around fixation control and object recognition," in *Proc. 2018 IEEE-RAS Int. Conf. Humanoid Robots*, 2018, pp. 840–845. doi: 10.1109/HUMANOIDS.2018.8625072.
[12] S. Makino et al., "Five-fingered hand with wide range of thumb using combination of machined springs and variable stiffness joints," in *Proc 2018 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 4562–4567. doi: 10.1109/IROS.2018.8594316.
[13] K. Shinjo et al., "Foot with a core-shell structural six-axis force sensor for pedal depressing and recovering from foot slipping during pedal pushing toward autonomous driving by humanoids," in *Proc. 2019 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 3049–3054. doi: 10.1109/IROS40897.2019.8967519.
[14] K. Kawaharazuka et al., "Long-time self-body image acquisition and its application to the control of musculoskeletal structures," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2965–2972, 2019. doi: 10.1109/LRA.2019.2923968.
[15] K. Kawaharazuka et al., "Task-specific self-body controller acquisition by musculoskeletal humanoids: Application to pedal control in autonomous driving," in *Proc. 2019 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 813–818. doi: 10.1109/IROS40897.2019.8967910.
[16] K. Kawaharazuka et al., "Reflex-based motion strategy of musculoskeletal humanoids under environmental contact using muscle relaxation control," in *Proc. 2019 IEEE-RAS Int. Conf. Humanoid Robots*, pp. 114–119. doi: 10.1109/Humanoids43949.2019.9034994.
[17] Y. Asano et al., "A sensor-driver integrated muscle module with high-tension measurability and flexibility for tendon-driven robots," in *Proc. 2015 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 5960–5965. doi: 10.1109/IROS.2015.7354225.
[18] K. Kawaharazuka, S. Makino, M. Kawamura, Y. Asano, Y. Kakiuchi, K. Okada, and M. Inaba, "Human mimetic forearm design with radioulnar joint using miniature bone-muscle modules and its applications," in *Proc. 2017 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 4956–4962. doi: 10.1109/IROS.2017.8206377.
[19] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986. doi: 10.1038/323533a0.
[20] J. Redmon and A. Farhadi, YOLOv3: An incremental improvement. 2018. [Online]. Available: arXiv:1804.02767

Kento Kawaharazuka, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: kawaharazuka@jsk.t.u-tokyo.ac.jp

Kei Tsuzuki, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: tsuzuki@jsk.t.u-tokyo.ac.jp

Yuya Koga, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: koga@jsk.t.u-tokyo.ac.jp

Yusuke Omura, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: omura@jsk.t.u-tokyo.ac.jp

Tasuku Makabe, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: makabe@jsk.t.u-tokyo.ac.jp

Koki Shinjo, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: shinjo@jsk.t.u-tokyo.ac.jp

Moritaka Onitsuka, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: onitsuka@jsk.t.u-tokyo.ac.jp

Yuya Nagamatsu, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: nagamatsu@jsk.t.u-tokyo.ac.jp

Yuki Asano, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: asano@jsk.t.u-tokyo.ac.jp

Kei Okada, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: okada@jsk.t.u-tokyo.ac.jp

Koji Kawasaki, Toyota Motor Corporation, Aichi, Japan. Email: koji_kawasaki@mail.toyota.co.jp

Masayuki Inaba, Department of Mechano-Informatics, Graduate School of Information Science and Technology, The University of Tokyo. Email: inaba@jsk.t.u-tokyo.ac.jp