INTECH
open science | open minds

OPEN ACCESS ARTICLE

International Journal of Advanced Robotic Systems

Framework for Creating Intuitive Motion Content for Humanoid Robots Based on Programming by Demonstration

Regular Paper

Gi il Kwon[1], Il Woong Jeong[2], Yong-Ho Seo[3,*] and Hyun Seung Yang[4]

1 The Robotics Program, Korea Advanced Institute of Science and Technology, Daejeon, Korea
2 Mobile Comm. Samsung Electronics Co., Ltd, Yeongtong-gu, Suwon, Republic of Korea
3 Department of Robot Engineering, Mokwon University, Daejeon, Korea
4 Department of Computer Science, Korea Advanced Institute of Science and Technology, Daejeon, Korea
* Corresponding author E-mail: yhseo@mokwon.ac.kr

Received 11 May 2012; Accepted 17 Aug 2012

DOI: 10.5772/52451

Abstract In this paper we propose an easy to use framework for creating motion content for humanoid robots based on user demonstrations. A wearable interface is proposed and a prototype of the wearable interface is implemented to provide an intuitive content creation method for users who lack expertise in robot programming. An algorithm for transferring human motion to a robot and creating editable motions is presented to improve the reusability of the demonstrated motions. For generating various motions from a limited number of demonstrations, two motion models are presented and diverse motions are generated from the motion models. A humanoid robot, AMIO, is used to validate the proposed motion generation framework.

Keywords Motion Generation, Humanoid Robots, Programming by Demonstration, Human-Robot Interaction

1. Introduction

A humanoid robot is a robot that has an appearance similar to that of humans. In addition to its appearance, the robot's abilities should also resemble those of humans so that it can provide the desired services in daily life. In the near future it will be possible to utilize humanoid robots for various entertainment purposes. For example, a robot will be able to play the role of a stage actor in a musical or take the place of a human player in a sports game. To enact such scenarios, humanoid robots should be able to generate various natural motions according to their intended purpose. However, today's humanoids are not capable of autonomous behaviour; in response, many researchers are studying how to program by demonstration, since imitation is often the most productive way to learn new skills.

One of the most important success factors for recent IT gadgets, such as smartphones, has been their ability to run diverse novel content. A humanoid robot should also be able to provide various content in order to succeed in the commercial market. In order to provide novel content for humanoid robots, an easy content creation tool is needed, especially for users with no expertise in robot programming.

Programming by Demonstration (PbD) is a robot-programming paradigm that teaches a robot new behaviours by demonstrating the task, instead of relying on machine level program codes. PbD research has primarily focused on motion primitive-based approaches. For example, Schaal et al. proposed a dynamic movement primitive to control the motors of a robot system [1]. Jenkins and Mataric generated new motions from action units extracted from motion capture data [2].

Another representative motion-generation method is the stochastic approach. Many researchers have shown interest in hidden Markov model-based representation and generation of motion, where motion is regarded as time-series data, similar to speech. Inamura et al. used a HMM to compress and decompress an original motion [3]. Calinon et al. encoded motions using PCA, ICA and HMM to generate motions for humanoid robots [4]. Kwon et al. tried to find the best combination of motion primitives to represent the demonstrated motion using a HMM [5].

Previous studies on motion generation based on programming by demonstration have, however, faced some limitations. The motion primitive-based approaches do not allow variations from the trained motions, and thus are not suitable for applications where diverse motions should be generated. The HMM-based motion generation approaches meanwhile require a great deal of data in the training process and much time in order to teach a robot a new motion. To overcome these limitations, a motion-generation method for humanoid robots based on user demonstration is proposed in this paper.

Our contributions are as follows: to provide an intuitive method for creating motion content for humanoid robots, the following features are presented. First, a wearable interface for capturing the user demonstration is designed and a prototype of the interface is developed. Two kinds of motion-editing techniques—direction transformation and magnitude scaling—are then proposed to provide variations of the original motion. Finally, two different motion regeneration methods are suggested to reflect the characteristics of the motion content. The overall flow of the content creation process is shown in Figure 1. By using our framework, a user can create editable motion and a method of modifying motion with diverse variation. Furthermore, our system does not need large training data to train robot motions.

This paper is organized as follows. In section 2 the motion capture process using our wearable interface is introduced. After explaining how to transfer human motion data to a humanoid robot in section 3, motion editing techniques are proposed in section 4. To represent the motion data for humanoid robots efficiently, two motion models are suggested in section 5. Two methods for generating new motions from the original motion sequence are then proposed in section 6. In section 7 the experimental results of our work are presented and we conclude our work in section 8.

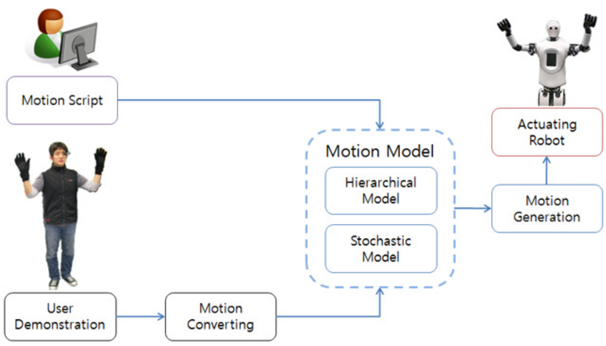


Figure 1. The overall motion content creation process.

2. A Wearable Interface for Motion Capture

Motion data is captured from a human demonstrator and then used to generate motions for humanoid robots. Since conventional motion capture devices tend to be very expensive and uncomfortable, a new wearable interface is proposed. The prototype is designed to capture the demonstrator's upper body motion, including motion of the head, both arms and both hands [6].

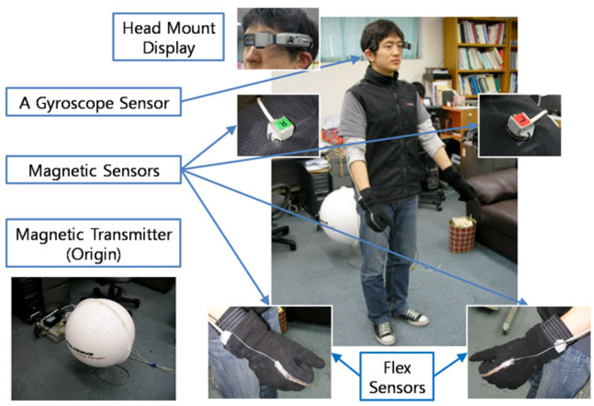


Figure 2. The prototype of the wearable interface.

A magnetic sensor system comprising four magnetic sensors is used to capture the motion of the arms. Each sensor can measure the 3-D position and 3-axis orientation angles from the origin transmitter. A 3-axis gyroscope sensor is attached at the rear of the head mount display. Two flex sensors that measure the bending angle are used to capture the motion of both hands. For this research, only the head mount display is used to show the internal state of the partner robot. The prototype of the wearable interface is shown in Figure 2.

There are two basic requirements for capturing the arm motion using the four magnetic sensors. First, the length of the demonstrator's arm must be given manually. With only two magnetic sensors, it is not possible to measure the distance from the shoulder joint to the elbow joint, and from the elbow joint to the wrist joint. Second, the demonstrator must perform a given motion only in a hemispheric shape in front of the sensor origin, as this is the valid workspace of the magnetic sensor. The positions of the wrist joint and the elbow joint are calculated by the following process, based on these two conditions.

We modelled the human arm as a 4 DOF arm. The shoulder has 3 DOF and the elbow has 1 DOF. We modelled the wrist as the end point of the arm. When the robot performs a motion, such as dancing, the wrist motion is rather small. Accordingly, we do not consider the wrist motion. Figure 3 shows a simplified model of the human arm.

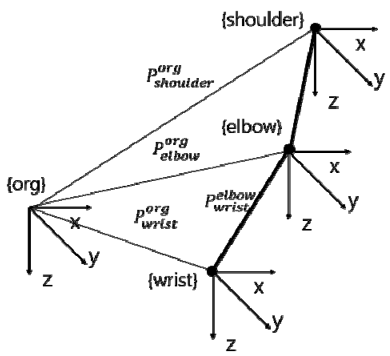


Figure 3. A simplified human arm model.

The position of the wrist joint P^{org}_{wrist} and the orientation of the wrist joint R^{org}_{wrist} are measured by a magnetic marker attached to the wrist joint. The position of the elbow joint at the initial posture P^{wrist}_{elbow} is defined as (1). The initial posture is standing to attention.

$$P^{wrist}_{elbow} = \begin{bmatrix} 0 \\ 0 \\ -ForearmLength \end{bmatrix} \quad (1)$$

The position of the elbow joint P^{org}_{elbow} in the global coordinate system of the magnetic sensor system can be calculated by (2), where R^{org}_{wrist} is a 3×3 transformation matrix that represents rotation of the wrist joint.

$$P^{org}_{elbow} = P^{org}_{wrist} - R^{org}_{wrist} P^{wrist}_{elbow} \quad (2)$$

The above process is conducted for both arms to calculate the motion data. The position of the wrist joints, elbow joints and shoulder joints, and the rotation angles of the neck joints from the 3-axis gyroscope sensor attached to the back of the neck are used as the motion data for the upper body movement. Figure 4 shows some motion frames captured using the proposed wearable interface. The arrows represent the position and the orientation of the magnetic markers attached to the important joints. The spheres indicate the calculated positions of the elbow and shoulder joints of both arms.

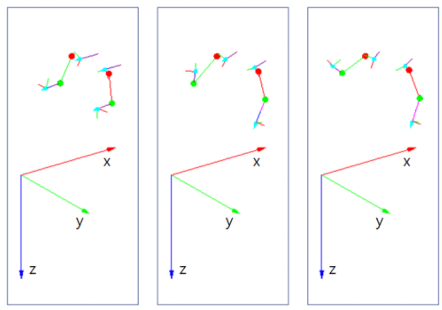


Figure 4. Some motion frames captured by the prototype of the wearable interface.

3. Transferring Human Motion to a Humanoid Robot

Because of the discordance between the arm structure of the demonstrator and that of the robot, it is necessary to convert the captured motion data. A normalization process and kinematic analysis are used to transform the captured motion into motion that is appropriate for the target robot. The normalization process corrects the difference between the arm length of the user and that of the robot, and the kinematic analysis compensates for the difference between the degrees of freedom of the user and the robot.

3.1 Normalization

The discordance in arm lengths between the demonstrator and the robot can be compensated for by a normalization process. The position of the demonstrator's wrist (in relation to his shoulder) is normalized by his arm length to compensate for the different segment length relative to the target robot. The normalized position of the wrist joint $P^{normalized}_{wrist}$ can be calculated by (3).

$$P^{normalized}_{wrist} = \frac{PH_{wrist} - PH_{shoulder}}{LH_{upper} + LH_{lower}}, \quad (3)$$

where PH_{wrist} and $PH_{shoulder}$ indicate the position of the demonstrator’s wrist joint and shoulder joint, and

LH_{upper} and LH_{lower} represent the length of the upper arm and lower arm, respectively. From $P^{normalized}_{wrist}$, the position of the wrist joint of the robot PR_{wrist} can be calculated by (4), where LR_{upper} and LR_{lower} represent the length of the upper arm and that of the lower arm of the robot, respectively.

$$PR_{wrist} = P^{normalized}_{wrist} \times (LR_{upper} + LR_{lower}) \qquad (4)$$

3.2 Kinematic Analysis

For use with an actual robot, the motion data should be converted into the robot's particular joint angle trajectories. A kinematic analysis is adopted to transform the human joint position trajectories into the angular trajectories of the robot. The 3-D positions of the three arm joints (shoulder, elbow and wrist) are used to solve the kinematic problem. The angular trajectories of the arm joints are calculated by solving the inverse kinematics problem. These trajectories are then used for the original motion sequence of the robot and the forward kinematics is calculated to validate the calculated joint angle trajectories.

4. Motion-Editing Techniques

Generating new motions by editing the original motion data is impractical since the data consist of dense sequences of postures (usually 30 postures per second), which are too numerous to modify. In order to make it easier to modify a given motion, key frame motion consisting of fewer motion frames is used [7].

4.1 Key Frame Extraction

Key frames are extracted from the original motion via a curve-simplification algorithm that is applied to each joint-angle trajectory. Curve simplification has been used in computer graphics, pattern recognition, image processing, computer vision and computational geometry in order to reduce the dimensionality of huge data sets. The curve-simplification algorithm proposed by Lowe is used to extract the key frame postures [8].

Each joint-angle trajectory for the robot can be represented as a curve. To simplify the original motion as a key frame motion for the humanoid robot, the curve-simplification algorithm is applied to each curve to extract the key frames. Figure 5 shows an example of simplified data for joint-angle trajectories of the left arm. The triangles represent the key frames in the joint-angle trajectories.

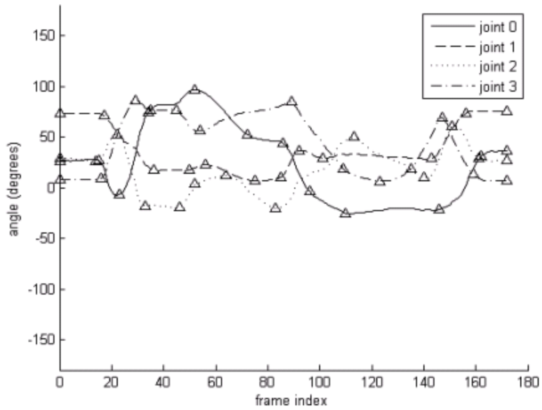


Figure 5. The simplified joint-angle trajectories of an arm for a certain motion.

Every motion frame that includes the simplified data is selected as a key frame candidate. Figure 6 shows all of the candidates for key frames in all of the joint-angle trajectories. Since there are many similar motion frames among the candidates, a projection such as this is necessary to select the important frames from the candidates.

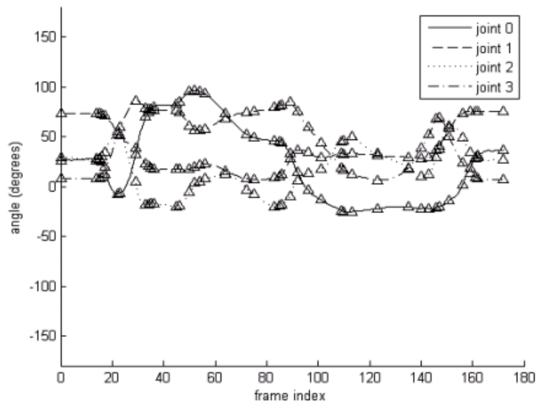


Figure 6. All key frame candidates are projected into each joint angle trajectory.

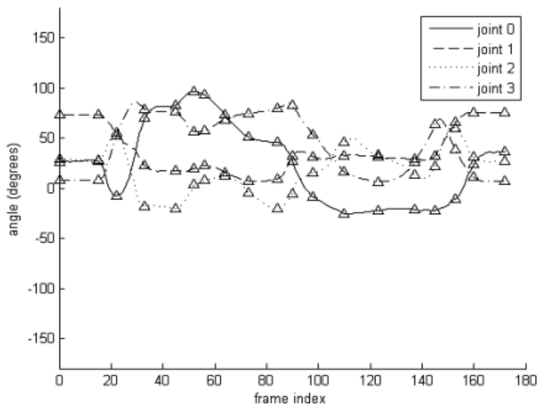


Figure 7. Precise key frames chosen by ART clustering.

A clustering algorithm is used to cluster the frame indices of the key frame candidates to create a more precise key frame motion, as shown in Figure 7. Because the number of key frames is unknown, k-Means clustering (which requires the exact number of clusters) is not applicable. Instead of k-Means clustering, Adaptive Resonance Theory (ART) is used to cluster the indices of the key frame candidates [9, 10, and 11].

4.2 Direction Transformation

The new 3-D position and 3-axis direction of the wrist joint in the shoulder coordinate system should be calculated in order to transform the direction of the motion. The 3-D position of the wrist joint $P_{wrist}^{rotated}$ can be calculated using a transformation matrix, which represents the rotation as delineated by (5), where R_{trans} represents the direction transformation matrix.

$$P_{wrist}^{rotated} = R_{trans} \times P_{wrist} \tag{5}$$

To apply the effect of the direction transformation to the wrist joint, both the 3-D position and 3-axis rotation angles of the wrist joint should be changed. The direction change should be applied to the rotation matrix using (6).

$$R_{wrist}^{rotated} = R_{trans} \times R_{wrist} \tag{6}$$

Extracting the exact rotation angles for each axis from the rotation matrix $R_{wrist}^{rotated}$ in (6) is impractical. Instead of calculating the direction of the wrist joint using (6), the direction transformed position of the elbow joint $P_{elbow}^{temp'}$ can be calculated by using (1) and (6).

$$P_{elbow}^{temp'} = R_{wrist}^{rotated} \times P_{elbow}^{0} \tag{7}$$

The position of the elbow joint $P_{elbow}^{rotated}$ can then be calculated by using (7).

$$P_{elbow}^{rotated} = P_{wrist}^{rotated} - P_{elbow}^{temp'} \tag{9}$$

Figure 8 shows the direction transformed motion generation process.

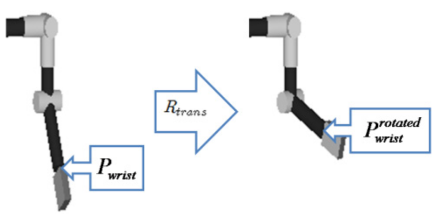


Figure 8. Direction transformed motion generation.

4.3 Magnitude Scaling

A motion plane is adopted to change the magnitude of the motion based on the wrist trajectories [12]. The motion plane is the plane that remains closest to every position of the wrist throughout the duration of the motion. The motion plane p can be represented by (10)

$$p : \omega_1 x + \omega_2 y + \omega_3 z + \omega_4 = 0 \tag{10}$$

The plane can be obtained by minimizing the sum of the distances between the plane and each wrist position, as shown in (11).

$$\begin{gathered} P_{wrist}(i) = (x_i, y_i, z_i) \\ d_i = \frac{|\,\omega_1 x_i + \omega_2 y_i + \omega_3 z_i + \omega_4\,|}{\sqrt{\omega_1^2 + \omega_2^2 + \omega_3^2}} \\ \underset{\Omega}{\arg\min} \sum d_i \text{ , where } \Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\} \end{gathered} \tag{11}$$

After calculating the motion plane, every wrist position is projected into the plane. The shape that represents the projected wrist positions can then be used to calculate the position of the centroid of the shape. By scaling the distance between the centroid and each projected wrist position, the projected wrist positions for the magnitude-changed motion can be calculated. The process of changing the scale of the magnitude is shown in Figure 9.

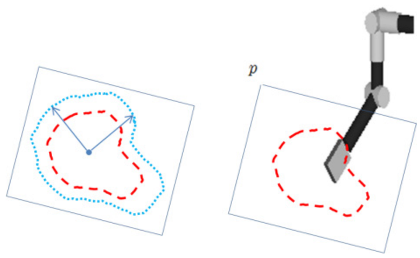


Figure 9. Magnitude scaling process.

After finding the scaled wrist trajectory in the motion plane, the positions of the wrist joint for the magnitude-changed motion can be calculated by projecting back from the original position of the wrist. The direction of the magnitude-scaled wrist is set to the direction of the new wrist position relative to the elbow position, when the wrist is at the centroid of the motion plane. The back projection process is shown in Figure 10.

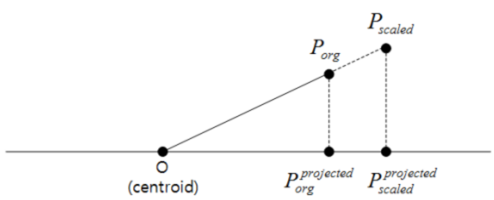


Figure 10. Calculation of the scaled position of the wrist joint.

5. Motion Models

This research considers two representation methods for expressing the motions of humanoid robots. The first method is a hierarchical motion model that represents a motion as a combination of atomic motions. In this model, the motions are classified into one of three classes: atomic actions, basic behaviours and task scenarios. A motion of the upper class consists of a combination of motions from the lower class. This method is suitable for expressing short motions for conducting a task. The second method is a stochastic motion model based on the original motion sequence. This method is appropriate for representing a prolonged motion with repetitive patterns.

5.1 Hierarchical Motion Model

Teaching a robot a new motion and its meaning is very difficult given the current intelligence levels of humanoid robots. Thus, instead of attempting to convey the meaning of the demonstrated motion, a hierarchical motion model that simply categorizes motions is proposed. The hierarchical motion model comprises three levels of motion hierarchy: atomic actions, basic behaviours and task scenarios.

An atomic action is a simple movement, such as raising the right arm or waving the left arm, as demonstrated by a user. The user should manually label the action after the demonstration. Atomic actions are the basic units of the motion hierarchy. After teaching a pool of atomic actions, the user can then define behaviours that consist of a combination of those atomic actions. Finally, a task scenario for the robot can be described through a composition of behaviours.

A new motion can be generated by a script for a task according to the hierarchical meaning of the motion. The details of the motion script based on the motion hierarchy will be explained in section 6.

5.2 Stochastic Motion Model

The hierarchical motion model is not suitable for representing certain motions that consist of repetitive patterns, such as dancing. In order to consider the variations of motions with repetitive patterns, a stochastic motion model is proposed.

To create a stochastic model for a motion, key frames should be extracted from the original motion. A clustering algorithm is then applied to the key frames to classify them into groups of similar postures. Since the number of clusters is unknown, an ART-based clustering (Gaussian ART clustering) is used to reflect the probabilistic characteristics of the motion [13].

After creating the information for each cluster, a stochastic model for the demonstrated motion can be created by rewriting the sequence of key frames as a sequence of key frame clusters. Using the motion represented in the sequence of key frame clusters, we can then calculate the transition probabilities for all clusters. The transitions can be represented as a weighted directed graph. We used the transition probability graph as a base model for the demonstrated motions. The stochastic motion modelling process is shown in Figure 11.

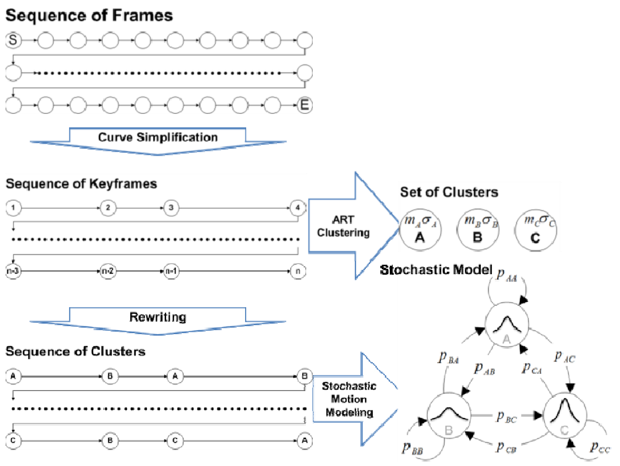


Figure 11. The process of generating the stochastic motion model.

6. Motion Regeneration

Two different methods are used to create new motions for humanoid robots from the motions captured from the human demonstrator. One is script-based motion generation, in which a motion is created according to a user-defined task scenario based on the motion hierarchy. The other method is stochastic motion generation. The created motion can be modified by the two transformation algorithms explained in section 3.

6.1 Hierarchical Motion Model

In order to create a motion from the meaning of the motion, the robot should be taught a series of simple actions (defined as combinations of atomic actions, as explained in section 5). An example of a hierarchical motion model is shown in Figure 12.

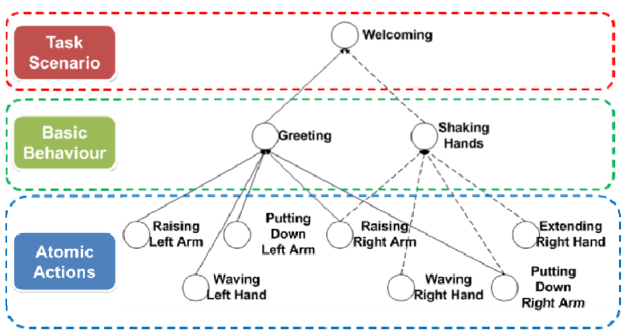


Figure 12. An example of a hierarchical motion model.

As shown in Figure 12, possible atomic actions include raising the left arm, lowering the left arm, waving the left arm and extending the right hand. After teaching the robot the atomic actions, higher level motions can be generated by combining the atomic actions. In the motion hierarchy, such higher level motions are defined as basic behaviours. For example, the greeting motion and the hand-shaking motion are basic behaviours that consist of combinations of atomic actions. A scenario for a task for the robot can then be created through a composition of the basic behaviours. For instance, the robot could learn a welcoming task comprising both the greeting and hand-shaking behaviour. A motion database for humanoid robots can be constructed by using the motion hierarchy. From the motion database, a scenario for a new task can be described by a script-based representation.

6.2 Stochastic Motion Model

A new motion is generated based on the stochastic motion model explained in section 5. During the motion-generation process, a key frame cluster is selected based on the transition probabilities of all of the clusters in the graph. A new key frame is generated from the selected key frame cluster based on the distribution of the selected cluster. After creating the first key frame, the next key frame is created by selecting the next cluster according to the transition probabilities of the first cluster. The above processes are repeated until the number of created key frames is sufficient to produce the proper motion. This process can generate motion with some variations, thus this method is viable for creating motions for entertainment use, such as dance and martial arts.

7. Experimental Results

7.1 Experimental Setup

We tested our motion generation method on AMIO, a bipedal humanoid robot developed in 2006 by Yang et al. [14]. AMIO is designed to test biped walking algorithms and to imitate natural human motions. With a built-in lithium-polymer battery, AMIO can operate for up to 30 minutes without an external source of electricity. AMIO is 150cm in height and 45kg in weight, and it can walk at speeds up to 1km/h. The appearance of AMIO and the number of degrees of freedom are shown in Figure 13.

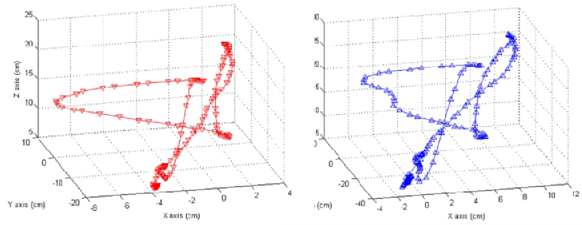
7.2 Motion Transferring

A heart-drawing motion is demonstrated by a human user. Figure 13 shows the trajectories of the left wrist joint of the demonstrator (Figure 14(a)) and AMIO (Figure. 14(b)).

Neck		2 DOF
Arms	Shoulder	3 DOF × 2
	Elbow	1 DOF × 2
	Wrist	1 DOF × 2
Hands		6 DOF × 2
Legs	Hip	3 DOF × 2
	Knee	1 DOF × 2
	Ankle	2 DOF × 2
Total		36 DOF

Degrees of freedom

Figure 13. The appearance and degrees of freedom of a humanoid, AMIO.



(a) Demonstrator's wrist trajectory (b) Wrist trajectory of the robot

Figure 14. The trajectories of the demonstrator's wrist joint and the robot's wrist joint.

Figure 15 show an example of the generated motion from Fig. 14. The original captured motion is shown in Fig. 15(a). Instead of applying the generated motion directly, the motion is validated through the 3-D robot model simulator, as shown in Fig. 15(b). After the safety of the generated motion is verified, the motion can be applied to the real robot platform. The wrist trajectory employed in drawing a heart is shown in Fig. 15(c).

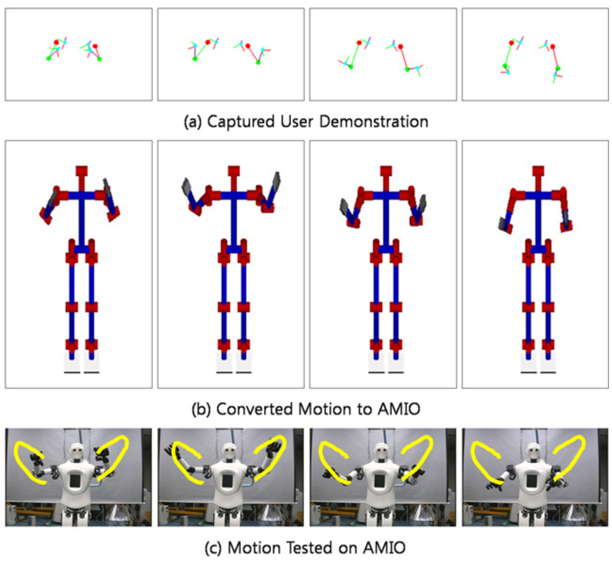


Figure 15. A heart-drawing motion and related test on a humanoid robot, AMIO.

7.3 Transformed Motion

To test the motion transformation algorithm, a drawing motion for the infinity symbol is used. The wrist trajectory of the demonstrator's arm is captured by the wearable interface and is shown in Figure 16. Each rectangle in Figure 16 represents the position of the wrist joint according to each key frame created by the key frame extraction method previously explained in section 4. Figure 17 and Figure 18 show the wrist trajectories of the direction-transformed motion and magnitude-scaled motion for the motion of drawing the infinity symbol.

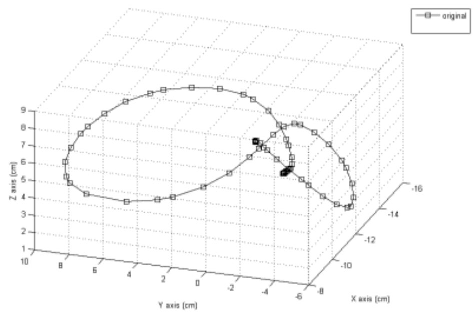


Figure 16. An infinity symbol-drawing motion.

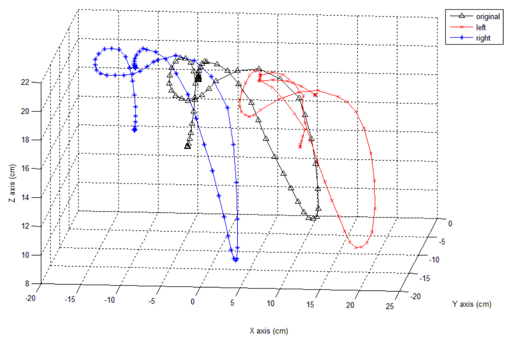


Figure 17. The direction-transformed motions.

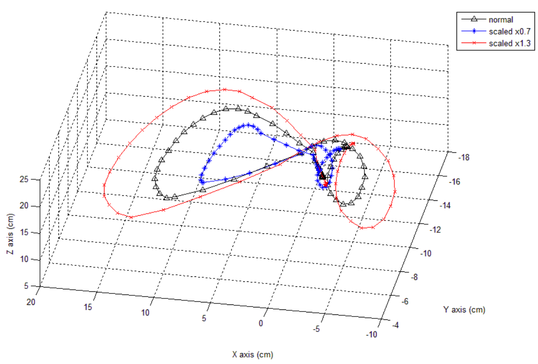


Figure 18. The magnitude-scaled motions.

7.4 Regenerated Motion

The backwards open-hand gesture commonly used to introduce a place serves as the example of a motion that is generated based on the hierarchical motion model. The motion consists of seven atomic actions and four behaviours as shown in Fig. 19. By combining all of the basic actions, this introducing motion can be generated by the humanoid robot. Motion-editing techniques described in section 4 can be used to vary the generated motion according to the purpose of the content.

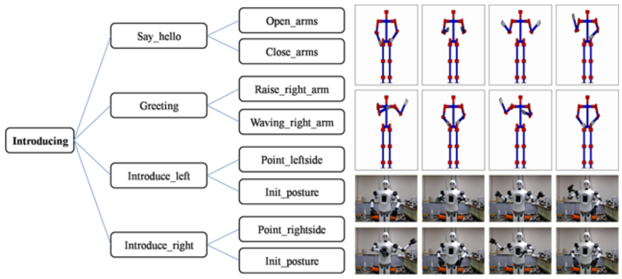


Figure 19. An example of a motion that is generated based on a hierarchical model.

A boxing motion consisting of simple boxing actions is used as an example of a motion generated from the stochastic motion model. The boxing motion consists of 4 basic actions: raising the arms to guard the face, jabbing with the left arm, punching straight with the right arm, and dropping guard. By combining these basic actions, an extended series of boxing motion for the humanoid robot can be generated. An example of the generated motion is shown in Figure 20.

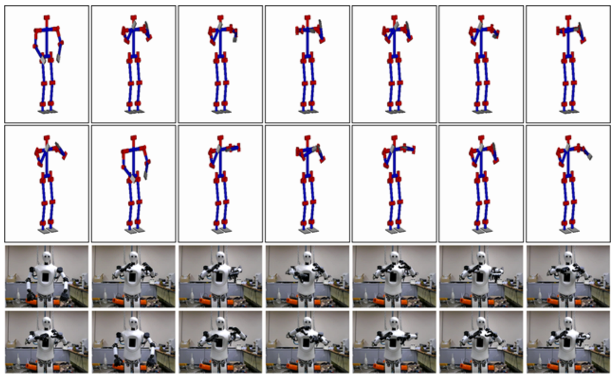


Figure 20. Boxing motion : An example of motion that is generated based on the stochastic motion model.

Unfortunately, it is difficult to measure the performance of the generated motion quantitatively due to the lack of common quantitative evaluation criteria. Instead of conducting quantitative evaluations, qualitative comparisons with the conventional system are used to show the advantages of this work. In terms of tracking precision, conventional systems surpass the proposed system, because the motion-capture devices used in conventional studies have high resolution of data acquisition. However, in terms of usability, the proposed system is superior to conventional systems since the proposed wearable interface is easy to wear and provides an intuitive method to generate new motions. Furthermore, various motions can be created by the motion-editing techniques proposed in this work. As a result, it is possible for a user who has no prior knowledge of robotics or programming to generate new motion content. The results of qualitative comparisons between conventional motion generation systems and the proposed system are shown in Figure 22.

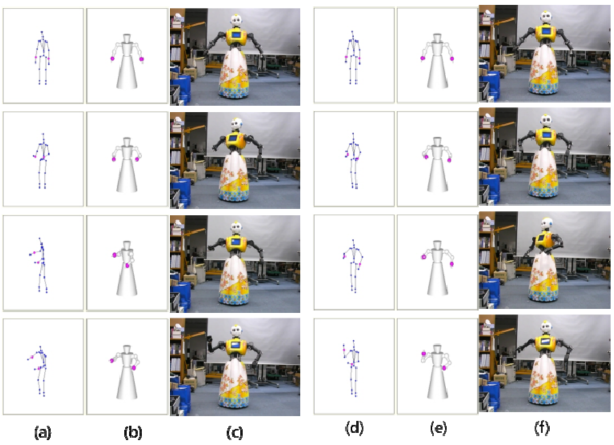


Figure 21. Dance motion: an example of motion generated from the stochastic motion model. (a),(d): the human motion captured by the motion capture interface, (b)(e): generated robot motion from capture data. (c)(f) : real robot motion.

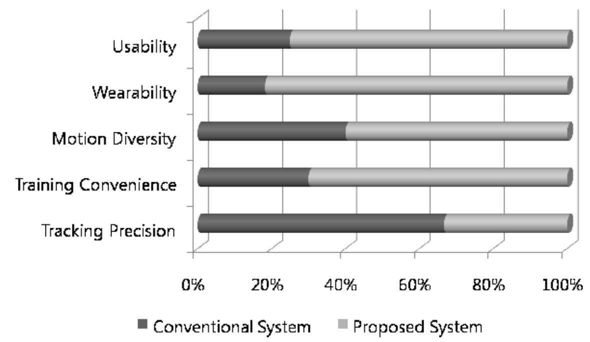


Figure 22. Qualitative comparisons with conventional approaches.

7. Conclusion

We proposed a framework that can be used to create novel motion content for humanoid robots with an intuitive interface based on user demonstrations. A wearable interface was designed and a prototype of the interface was developed to capture the human demonstration. A humanoid robot named AMIO was used to test the generated motions. We proposed a method of transferring human motion to humanoid robots, a method of creating editable motion, and a method of modifying motions with diverse variations. As a result, even users with no expertise in robot programming can easily create new motions for humanoid robots using the proposed system. It is anticipated that the novel motion content that can be created with this framework can revitalize the depressed market for service robots.

For further work, we intend to develop a system that generates new motions for interacting with humans from a pair of demonstrations by multiple users. We are also planning to create a hierarchical motion model that automatically recognizes the meaning of the demonstrated motion in order to respond to the user's intent without manual interpretation. Finally, another of our goals for future research is to establish quantitative evaluation criteria for a performance evaluation of the generated motions.

8. Acknowledgments

This research was supported by the Basic Science Research Program through a grant from the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0018291 and 2011-0013776). This work was also supported by the IT R&D program of MKE/MCST/IITA, [Development of learner-participational and interactive 3D virtual learning contents technology].

9. References

[1] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert (2003) Learning Movement Primitives. International Symposium on Robotic Research. pp. 561-572.

[2] O. C. Jenkins and M. J. Mataric (2002) Deriving Action and Behavior Primitives from Human Motion Data. International Conference on Intelligent Robots and Systems. pp.2551-2556.

[3] T. Inamura, H. Tanie, and Y. Nakamura (2003) Keyframe Compression and Decompression for Time Series Data based on the Continuous Hidden Markov Model. International Conference on Intelligent Robots and Systems. pp.1487-1492.

[4] S. Calinon and A. Bilard (2005) Recognition and Reproduction of Gestures using a Probabilistic Framework combining PCA, ICA and HMM. International Conference on Machine Learning. pp.105-112

[5] J. Kwon and F. C. Park (2008) Natural Movement Generation Using Hidden Markov Models and Principal Components. IEEE Transactions on Systems, Man, and Cybernetics – Part B : Cybernetics, vol. 38, no. 5, pp. 1990-1995.

[6] Y. Seo, I. Jeong, and H. Yang(2007) Motion capture-based wearable interaction system and its application to a humanoid robot, AMIO. Advanced Robotics, vol. 21, no. 15, pp.1725-1741.

[7] I. Lim and D. Thalmann (2001) Key-posture extraction out of human motion data by curve

simplification. International Conference of the IEEE Engineering in Medicine and Biology Society. vol.2, pp.1167-1169.
[8] D. Lowe (1987) Three-dimensional object recognition from single two-dimensional images. Artificial Intelligence, vol. 31, pp.355-395.
[9] A. Baraldi and E. Alpaydin (2002) Constructive Feedforward ART Clustering Networks- Part I. IEEE Transaction on Neural Networks, vol. 13, no. 3, pp. 645-661.
[10] A. Baraldi and E. Alpaydin (2002) Constructive Feedforward ART Clustering Networks-Part II. IEEE Transactions on Neural Networks, vol. 13, no. 3, pp. 645-661.
[11] E. Alpaydin (2004) Introduction to Machine Learning. The MIT Press : Cambridge. pp.275-303
[12] C. Kim, S. Kim, S. Ra, and B. You (2007) Regenerating Human-like Arm Motions of Humanoid robots for a Movable Object. SICE Annual Conference. pp.1081-1086.
[13] J. R. Williamson (1996) Gaussian ARTMAP : A Neural Network for Fast Incremental Learning of Noisy Multidimensional Maps. Neural Networks, vol.9, no. 5, pp. 881-897.
[14] H. Yang, Y. Seo, Y. Chae, I. Jeong, W. Kang, and J. Lee (2006) Design and Development of a Biped Humanoid Robot, AMI2, for Social Interaction with Humans. IEEE-RAS International Conference on Humanoid Robots, pp.352-357.