2005 IEEE/RSJ International Conference on Intelligent Robots and Systems

Task Model of Lower Body Motion for a Biped Humanoid Robot to Imitate Human Dances

Shinichiro Nakaoka*, Atsushi Nakazawa†, Fumio Kanehiro‡, Kenji Kaneko‡,
Mitsuharu Morisawa‡ and Katsushi Ikeuchi*

*Institute of Industrial Science, University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo, JAPAN. {nakaoka, ki}@cvl.iis.u-tokyo.ac.jp
†Cyber Media Center, Osaka University
1-21 Machikaneyama, Toyonaka, Osaka, JAPAN. nakazawa@ime.cmc.osaka-u.ac.jp
‡Intelligent Systems Institute, National Institute of Advanced Industrial Science and Technology
Tsukuba Central 2, 1-1-1 Umezono, Tsukuba, Ibaraki, JAPAN. {f.kenehiro, k.kaneko, m.morisawa}@aist.go.jp

***Abstract*— The goal of this study is developing a biped humanoid robot that can observe a human dance performance and imitate it. To achieve this goal, we propose a *task model* of lower body motion, which consists of *task primitives* (what to do) and *skill parameters* (how to do it). Based on this model, a sequence of task primitives and their skill parameters are detected from human motion, and robot motion is regenerated from the detected result under constraints of a robot. This model can generate human-like lower body motion including various waist motions as well as various stepping motions of the legs. Generated motions can be performed stably on an actual robot supported by its own legs. We used improved robot hardware *HRP-2*, which has superior features in body weight, actuators, and DOF of the waist. By using the proposed method and HRP-2, we have realized a dance performance of Japanese folk dance by the robot, which is synchronized with a performance of a human grand master on the same stage.**

***Index Terms*— biped humanoid robot, human motion, imitation learning, dance, motion capturing system**

I. Introduction

The goal of this study is developing a biped humanoid robot that can observe a human dance performance and imitate it. The main purpose of developing such a robot is to preserve traditional dances that are considered important intangible cultural heritages. Many of those dances are disappearing because there are few successors. Our idea is to make a robot master those dances and perform them again as a successor of the dances in the real world. This kind of preservation will be more effective than video recordings or 3D computer graphics. No matter how much those media can express reality, a real robot is the best medium in reality if the robot can perform dances like a human dancer.

It is a valuable application of technology to enable a humanoid robot to perform motions that require a human-like body. Dance is one of these applications, and there have been a number of studies related to a dancing humanoid ([1] [2] [3] [4] [5] [6]).

Some studies use a motion-capturing system to acquire the original motion. Pollard et al. [1] proposed a method for importing captured human motion into a humanoid robot. The robot they used is fixed to a stand at the waist and their method is available for generating upper body motion. Yamane et al. [2] proposed a method of controlling a marionette according to a captured human motion. These studies solve a problem of importing human motion into a humanoid that does not have exactly the same body structure as that of humans. However, the humanoids that are used in the above studies are not a biped walking robot. Importing leg motion is an especially difficult problem because it must support the whole body and consider body balance under constraints caused by rigid soles and actuators' limit, etc.

Sony [3] has developed a small biped humanoid robot and has realized a dance performance using it. Although its performance is dynamic motion including legs, the motion is created by a human programmer so that the motion can be adapted to the robot hardware [7].

As compared with related works, a novel achievement of this study is realizing two key features. (1) The motions of the robot are not originally designed for the robot, but are imitative motions based on a human dance performance. (2) We use a robot that is a human-size biped walking robot and is supported by its own legs.

Our solution to achieve these features is based on the paradigm of *Learning from Observation (LFO)*[8][9]. In this paradigm, *task primitives* (what to do) and *skill parameters* of them (how to do it) are defined as a task model. Then a robot extracts tasks and skills from human motion by observing it, and they become commands to move the robot. We had already proposed a task model of lower body motion to realize dynamic whole body motions such as dances [10]. As for lower body motion, the paradigm that robot motion is regenerated from the recognized tasks under constraints of a robot works more effectively than adapting the original form of motion data to the robot. In this paper, we improve our previous model so that it supports various waist motions in order to express more human-like motions. The method for

0-7803-8912-3/05/$20.00

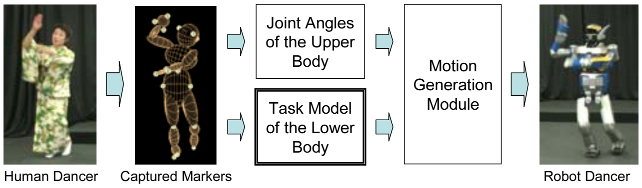

Fig. 1. Overview process

generating robot motion based on the model is also improved in two respects. Foot trajectory is generated so that impact force at touching down is reduced, and slipping of the soles is prevented by yaw moment compensation.

Hardware performance of a robot is also important in reproducing dynamic dance motions. In this paper, we use an improved humanoid robot *HRP-2* [11], which replaced *HRP-1S* we used in our past papers. HRP-2 has a lighter (58 k.g.) body and faster actuators so that it can move quickly and smoothly. It also has a two-DOF joint (pitch and yaw) between the waist and the chest so that it can express more complex motions.

As the final result, we have realized an imitative dance dance performance by HRP-2, which is synchronized with the original music and a human grand master on the same stage.

II. Overview

Fig. 1 shows the overview process of this study.

First, a human dancer dances and his/her motion is acquired as position sequences of a number of body parts. It is desirable that motion is acquired by means of vision on a robot, but currently we have used a motion-capturing system in this process.

Then lower body motion are recognized as a sequence of task primitives based on the task model. The task model and its recognition process is described in section III. On the other hand, upper body motion is converted into joint angle sequences which can be input to a robot. This process is done by inverse kinematics (IK) of upper body parts and some techniques for adapting joint angle sequences to joint constraints. Details of this process are described in [1][12].

Both the recognized primitives and joint angles of the upper body are input into a *motion generation module*. It generates joint angle sequences of the whole body, which satisfy dynamics and constraints of the robot. Details of the generation process is described in section IV.

Finally, the generated motion data is input into the robot, and the robot performs an imitative dance motion. We show experimental results on HRP-2 in section V.

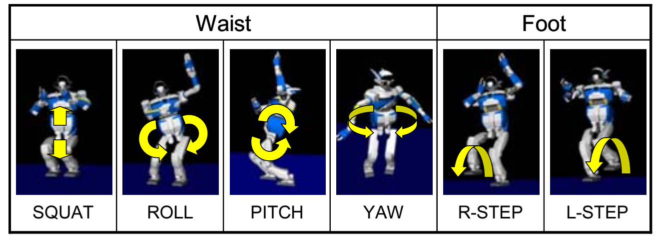

Fig. 2. Task primitives

TABLE I
SKILL PARAMETERS

	Key States	Parameters
Common	Initial state, final state	Beginning time (initial state), duration time
SQUAT	Medium state: The waist reaches the lowest position	Waist height distance between the initial state and the medium state
ROLL, PITCH, YAW	-	Destination angle at the final state
STEP	Medium state: A swinging foot reaches the highest position	The Position and the attitude of the swing foot at the medium state and the final state. These values are described in relative coordinate from the supporting foot.

III. Task model of lower body motion

A. Task primitives

A whole sequence of the original motion is decomposed into a number of task primitives. Task primitives are defined for waist motion and foot motion as fig. 2. Each primitive has a time span and is put on the time axis. The same type of primitive does not simultaneously appear, and different kinds of primitives can be concurrent.

Horizontal translations of the waist are not defined, because they are strongly affected by dynamic balance maintenance. Details of this are described in section IV-E.

Although the task model itself can express motions such as jumping and running by overlapping R-STEP and L-STEP, our method has not supported generating those motions so far. In fact, running has been realized by Kajita et al. [13] on *HRP-2LR*. However, to perform typical jumping and running that are found in dances has still been difficult for the current hardware.

B. Skill parameters

A task primitive has its own skill parameters to express its execution time span and motion characteristics. A parameter set basically consists of times and positions at *key states*. Details of skill parameters are shown in table I.

All the primitives can take any position at the initial state, which is inherited from the final state of the previous motion. Parameters of a swinging foot in STEP are described as relative values from the supporting foot. In this way, each primitive is independent of other primitives and parameters can be modified independently. This feature is necessary in a refinement process described in section IV-C.

C. Recognizing tasks from captured human motions

Task primitives and their skill parameters are extracted from captured motion data. First, segments (including key frames) of primitives are detected, then position parameters are extracted.

Primitive segments are detected by analyzing velocity (or speed) of a body part related to a target primitive. For example, Fig. 3-(a) shows a speed graph of a foot. By analyzing this kind of a graph, segments of step primitives can be detected. When a foot is stepping, it increases its speed, then reduces speed and stops. That is, a convex trajectory appears in the graph when the foot is stepping. Therefore, by detecting those trajectories, segments of stepping are acquired. In this process, a threshold value in terms of moving path distance (integration of speed) should be defined, and segments that do not satisfy the threshold should be eliminated, because those segments are just a noisy movement.

As for waist primitives, velocities of the height, the roll angle, the pitch angle and the yaw angle are analyzed independently. These elements correspond with primitives of SQUAT, ROLL, PITCH and YAW respectively. For example, fig. 3-(b) shows a velocity graph of the waist height. In this graph, a convex curve delimited by the time-axis is considered one vertical movement of the waist. One squatting motion corresponds to a pair of curves which is marked out in the graph. In this way, segments of SQUAT can be detected. Primitives of ROLL, PITCH and YAW can be also detected by analyzing an angular velocity of the waist attitude in a similar way. In these primitives, one curve delimited by the time-axis corresponds to one segment. In the detection, thresholds for eliminating noisy movements are defined as well as STEP primitive.

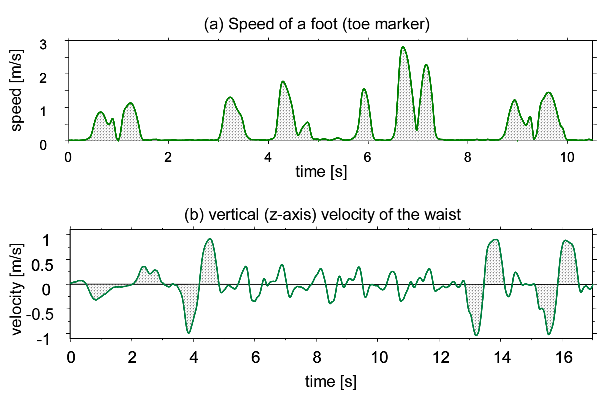

Fig. 3. Graphs for detecting primitive segments. Filled areas show segments which satisfy the threshold of detection.

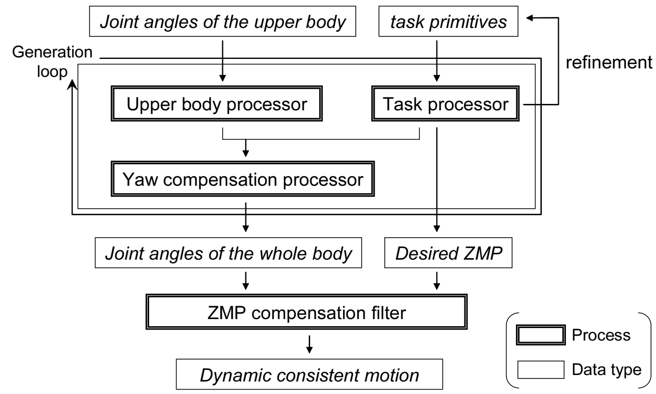

Fig. 4. Overview process in the motion generation module

After each segment of a primitive is detected, position parameters of it are extracted from the motion data. For example, positions and attitude of a swing foot in a STEP primitive are extracted from the captured marker positions of the both feet at frames of key states.

IV. Generating robot motion

A. Overview process

Fig. 4 shows an overview of the motion generation module. Joint angles of the whole body are generated by a generation loop that corresponds a discrete time frequency. There are three *processors* called *upper body processor*, *task processor* and *yaw compensation processor*. The upper body processor just sets joint angles of the upper body which is described in section II. The other processors are described in the following sections.

In the generation loop, if problems such as self-collisions of the feet are detected, the task processor modifies skill parameters of related primitives to eliminate the problems. This process is called *refinement of skill parameters*.

After the whole motion sequence is generated, *ZMP compensation filter* is applied to the generated data. This filter modifies the horizontal position of the waist according to *desired ZMP* generated by the task processor so that the motion satisfies dynamics consistency.

B. Task processor

Task processor is a main part of the generation process. It generates joint angles of the lower body according to input task primitives. In one generation loop, the following process is performed. First, the position and attitude of the waist and the feet are set in order to realize motions corresponding to the task primitives. Then joint angles of legs are calculated by IK between the waist and a foot.

Motion trajectories of the waist and the feet are generated by *primitive generators* in the task processor. A generator is awaked when a corresponding primitive begins to be performed, and the generator manages body properties related to the primitive until the primitive has finished its performance. There is a specialized generator for each primitive, and each generator works independently.

A primitive generator for SQUAT (*SQUAT generator*) set the waist height according to SQUAT primitives. Primitive generators for ROLL, PITCH and YAW set the waist roll-pitch-yaw angles respectively according to those primitives. Since a primitive is expressed by skill parameters of a few key states, a generator generates whole the trajectory of corresponding properties by interpolating height values or angle values of the states. Current implementation generates a trajectory by a cubic spline. One segment of the spline corresponds one segment between two key states. If additional waist joints such as a yaw joint between the waist and the chest are available, these generators can also set angles of such joints.

A primitive generator for a STEP primitive (*STEP generator*) generates trajectories of the feet (with respect to position and attitude) in a similar way of waist task primitives, but the foot trajectory has to consider the following requirements.

First, the trajectory has to satisfy the contact condition with a floor. A sole of the current robot is not so flexible as that of humans, but a rigid plate that must contact to be completely flat with the floor. This problem is easily solved by forcing the position and the attitude parameters of the swinging foot at the final state to be level with the floor.

Second, the reaction force impact from the floor has to be reduced as much as possible in order to step stably. To achieve this, we define *horizontal smoothness factor* and *vertical smoothness factor* in generating a swing foot trajectory. The former defines a height parameter Z_h, and horizontal (x-y position) movement of a swinging foot is finished before the foot reaches below Z_h. This kind of a trajectory can reduce a horizontal impact at touching down. The latter defines a height parameter Z_v and vertical velocity parameter V_v, and vertical velocity of a swinging foot is decreased to V_v before the foot reaches below Z_v by appending an additional segment to the spline of the height (z position) trajectory. This kind of a trajectory can reduce a vertical impact at touching down.

C. Refinement of skill parameters

Although a humanoid body is designed to be similar to a human body, they are not completely the same. Therefore, in some cases, skill parameters that are extracted from human motion cannot be directly performed on a robot. For example, fig. 5 shows a case of a self collision between the feet. This kind of problems are solved by modifying skill parameters slightly. Since the original parameters were actually performed by a human, the parameters are not so different from the desired values on the robot. In this case, a position of the final state in a STEP primitive is translated to the position that does not cause the collision by extending the distance between the feet. Then the generation loop goes back to the time when the related primitives had not been performed, and retries generation again.

Currently, automatic parameter refinement is implemented only in this case (avoiding collisions of the feet).

D. Yaw compensation processor

If the yaw moment in which soles put on the floor is above the friction force, the soles slips. This problem should be prevented. Tamiya et al. [14] proposed a method in which a robot maintains its body balance including yaw moment when it is being supported by only one leg. We improved a part of yaw moment compensation in this method so that it can accommodate any pattern of support by legs and can utilize the waist yaw joint of HRP-2. After the other processors are processed, their output is given to the yaw compensation processor. Then calculated magnitude of the yaw moment that soles put on the floor is constrained by controlling the waist yaw joint in order to prevent the slip. In other words, horizontal dynamics of the motion becomes consistent. The processor has a parameter of a friction factor μ. A proper value of μ has to be determined by experiments. (It is determined from the actual friction property between a robot and a floor.)

E. ZMP compensation filter and desired ZMP

Motion data generated by the above process does not necessarily satisfy vertical dynamics consistency between a robot and a floor, and the robot will not be able to maintain its balance and will fall down. ZMP has to be considered for this problem. If the motion data that includes global position and attitude in addition to all the joint angles is given, a *pseudo ZMP* can be calculated. In some cases, the pseudo ZMP is outside of the *supporting area*, which is formed by both the soles. This means that the motion does not satisfy the dynamics consistency. In other words, the motion must

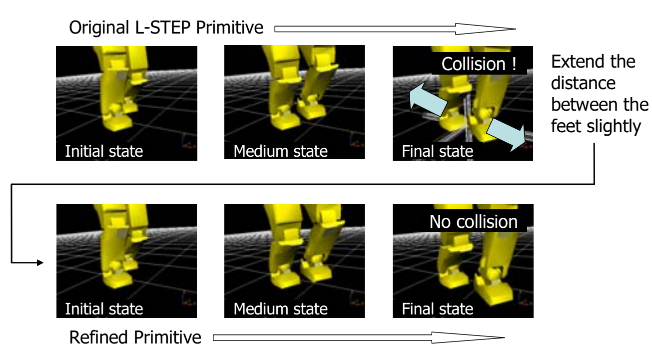

Fig. 5. Collisions between the feet are eliminated by refinement process

be modified so that a pseudo ZMP may always be inside the supporting area.

This problem can be solved by the method proposed by Nishiwaki et al. [15]. This method requires a desired ZMP trajectory and modify the original motion in a way that the upper body is horizontally translated so that a pseudo ZMP follows the desired ZMP. Since translating the upper body is approximation of translating the whole body, applying this method must be iterated until translation converges.

Our problem in using this method is how to make a desired ZMP trajectory. A robot takes one of three *support states*, which is fundamental to determine ZMP. When a robot is supported by the left foot, ZMP must be under the left sole. Same applies to the right foot. When a robot is supported by the both feet, ZMP must be inside the supporting area. In this case, the following criteria are desirable for stability: ZMP should smoothly move to the position required by the next support state, and ZMP should be center of the supporting area. Since support states are clearly determined by STEP primitives, the task processor can generate a desired ZMP trajectory based on the above criteria.

V. EXPERIMENTAL RESULT

A. Capturing a human dance motion

We chose a folk dance called 'Aizu-Bandaisan' as a subject of imitative performance by a robot. It has been inherited locally in Aizu, Japan. Although the motion of this dance is relatively slow and smooth, it has wide and dynamic motion using the whole body, and individual characteristics by a dancer appear well. This dance does not include jumping and running, which are difficult for the current robot. The included features are proper for the first trial.

We used *VICON* [16], which is an optical type motion-capturing system, in order to acquire human motion data. The system we used consists of eight-cameras and 32 markers that are attached to the human body. The system can acquire 3D positions of markers at the rate of 120 frames/sec. These specifications are sufficient for acquiring in-depth motion of the dance performance. By using this system, we captured a dance motion performed by a human grand master.

B. Result of Task Detection

We applied the method of detecting task primitives to the captured motion data. The method worked well: 24 R-STEPs, 19 L-STEPs and 7 SQUATs were detected from the performance of 35 sec. The result almost corresponds to recognition by a human. In this detection, the threshold of STEP is set to 0.1[m], and the threshold of SQUAT is set to 0.2[m].

C. Generating motion data of HRP-2

We generated motion data of HRP-2 according to the result of task detection. Basically the generation method worked well, but modification of the original skill parameters were necessary to prevent collisions between the feet and overruns of actuator's speed limit. Collisions between the feet were automatically eliminated by enabling the refinement process. Overruns of actuator's speed limits had to be eliminated by shortening stepping distances manually. Overruns occurred in 7 STEPs of all the 43 STEPs only on the knee joints, which has to move quickly to move a foot in postures where the knee joint is relatively extended (close to the singular point). In most refinements of STEPs, amounts of modification were below 5 cm, which can be considered as small modifications. To make this kind of refinement automatic is our future work.

D. Dance performance by HRP-2

HRP-2 was controlled by a controller with a stabilizing system [17]. In addition to joint angle sequences, sequences of desired ZMP and the waist attitude are given to the controller. Those data is used for stabilizing the robot according to sensor feedback in order that actual ZMP follows the desired position. Although the motion data satisfies theoretical ZMP condition, the stabilizing system is necessary due to disturbances and errors in theoretical model specifications.

Parameters of the processors are important for stable performance and are determined through experiments on the actual robot. The final (minimum) parameter values which are sufficient for generating stable motion are as follows: Z_h is $0.006[m]$, Z_v is $0.005[m]$, V_v is $0.13[m/s]$, μ is 0.25. Smoothness factors of the STEP generator worked effectively. When they were disabled, the whole body of the robot frequently quaked at the finish of stepping. However, that behavior was resolved by enabling them with proper values. The yaw compensation controller also worked well. μ seems a small value, which is due to a situation that part of a sole floated slightly above the floor when large yaw moment was on the sole. Actual friction of the sole in this case was very small, and μ became a small value in order to prevent slipping even in this situation.

Fig. 6 shows the resulting dance performance by HRP-2. HRP-2 successfully performed the dance synchronized with the original music and a performance by the grand master on the same stage [18].

E. Evaluation

As shown in fig. 6, a good result was obtained. Although the design of the task model is simple enough, it seems that characteristics of the dance motion are well expressed. The grand master commented that the robot could express the characteristics of her motion well, especially in leg motions. So our design of the task model is favorably evaluated by the grand master. However, some kind of theoretical criteria for evaluating similarity of dance motions should be defined. This is our future work.

Fig. 6. Performance of Aizu Bandaisan dance by HRP-2 and a grand master on the same stage.

VI. Conclusion

In this paper, we proposed the task model of lower body motion for a robot to imitate human dances. Based on this model, task primitives of lower body motion were correctly detected from motion data captured from human dances. We used the humanoid robot HRP-2, which has superior features in body weight, actuators, and DOF. The motion of HRP-2 was generated from the detected task primitives, considering constraints of the robot. We have realized a imitative dance performance of Japanese folk dance by HRP-2, which is synchronized with the original human speed. It is confirmed by the resulting dance performance that the model can generate natural human-like motion, including motion of the legs, under constraints of the current robot.

In future work, we will test dances by different dancers and also test other kinds of dances. Through those experiments, we will improve the model so that it may support more various expressions.

Acknowledgment

This work is supported in part by the Japan Science and Technology Agency (JST) under the Ikeuchi CREST project, and in part by the Grant-in-Aid for Scientific Research on Priority Areas (C) 16016218 of the Ministry of Education, Culture, Sports, Science and Technology.

We captured dance motions performed by the members of the dance group 'Aizu Gyokusui-kai'.

References

[1] N. S. Pollard, J. K.Hodgins, M. J. Riley, and C. G. Atkeson, "Adapting human motion for the control of a humanoid robot," in *Proceedings of International Conference on Robotics and Automation*, Washington DC, April 2002.

[2] K. YAMANE, J. K. Hodgins, and H. B. Brown, "Controlling a marionette with human motion capture data," in *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, September 2003, pp. 3834–3841.

[3] Y. Kuroki, M. Fujita, T. Ishida, K. Nagasaka, and J. Yamaguchi, "A small biped entertainment robot exploring attractive applications," in *Proceedings of IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, September 2003.

[4] R. Ikeura, H. Nakazato, and H. Inooka, "Motion planning for dancing robots based on human's dancing motion (in japanese)," vol. 15, no. 6, 1997, pp. 117–123.

[5] K. Kosuge, T. Hayashi, Y. Hirata, and R. Tobiyama, "Dance partner robot -Ms DancerR-," in *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, October 2004, pp. 3459–3464.

[6] X. ZHAO, Q. HUANG, Z. PENG, and K. LI, "Kinematics mapping and similarity evaluation of humanoid motion based on human motion capture," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, September 2004.

[7] Y. Kuroki *et al.*, "Motion creating system for a small biped entertainment robot," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, October 2003.

[8] K. Ikeuchi and T. Suehiro, "Toward an assembly plan from observation, Part I: Task recognition with polyhedral objects," *IEEE Transaction on Robotics and Automation*, vol. 10, no. 3, pp. 368–385, Junuary 1994.

[9] Y. Kuniyoshi, M. Inaba, and H. Inoue, "Learning by watching: Extracting reusable task knowledge from visual observation of human performance," *IEEE Transaction on Robotics and Automation*, vol. 10, no. 6, pp. 799–822, December 1994.

[10] S. Nakaoka, A. Nakazawa, K. Yokoi, and K. Ikeuchi, "Leg motion primitives for a dancing humanoid robot," in *Proceedings of IEEE International Conference on Robotics and Automation*, New Orleans, LA, USA, April 2004.

[11] K.Kaneko *et al.*, "Humanoid robot HRP-2," in *Proceedings of IEEE International Conference on Robotics and Automation*, New Orleans, LA, USA, April 2004.

[12] S. Nakaoka, A. Nakazawa, K. Yokoi, H. Hirukawa, and K. Ikeuchi, "Generating whole body motions for a biped humanoid robot from captured human dances," in *Proceedings of IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, September 2003.

[13] S. Kajita, T. Nagasaki, K. Kaneko, K. Yokoi, and K. Tanie, "A hop towards running humanoid robot," in *Proceedings of IEEE International Conference on Robotics and Automation*, New Orleans, LA, April 2004.

[14] Y. Tamiya, M. Inaba, and H. Inoue, "Realtime balance compensation for dynamic motion of full-body humanoid standing on one leg (in japanese)," *Journal of the Robotics Society of Japan*, vol. 17, no. 2, pp. 268–274, 1999.

[15] K. NISHIWAKI, S. KAGAMI, Y. Kuniyoshi, M. INABA, and H. INOUE, "Online generation of humanoid walking motion based on a fast generation method of motion pattern that follows desired zmp," in *Proceedings of International Conference on Intelligent Robots and Systems*, EPFL, Lausanne, Swizerland, October 2002.

[16] http://www.vicon.com/.

[17] K. KANEKO *et al.*, "Stabilizer for biped walking (in japanese)," in *Proceedings of the 21st Annual Conference of The Robotics Society of Japan*, Tokyo, Japan, September 2003.

[18] http://www.cvl.iis.u-tokyo.ac.jp/video.html.