Human robot interaction by understanding upper body gestures

Xiao, Yang; Zhang, Zhijun; Beck, Aryel; Yuan, Junsong; Thalmann, Daniel

2014

Xiao, Y., Zhang, Z., Beck, A., Yuan, J., & Thalmann, D. (2014). Human robot interaction by understanding upper body gestures. Presence : teleoperators and virtual environments, 23(2), 133-154.

https://hdl.handle.net/10356/100584

https://doi.org/10.1162/PRES_a_00176

Human-Robot Interaction by Understanding Upper Body Gestures

Yang Xiao, Zhijun Zhang*, Aryel Beck

Institute for Media Innovation, Nanyang Technological University, Singapore

Junsong Yuan

School of Electrlical and Electronic Engineering, Nanyang Technological University, Singapore

Daniel Thalmann

Institute for Media Innovation, Nanyang Technological University, Singapore

Abstract

In this paper, a human-robot interaction system based on a novel combination of sensors is proposed. It allows one person to interact with a humanoid social robot using natural body language. The robot understands the meaning of human upper body gestures and expresses itself by using a combination of body movements, facial expressions and verbal language. A set of 12 upper body gestures is involved for communication. This set also includes gestures with human-object interactions. The gestures are characterized by the head, arm and hand posture information. The wearable Immersion CyberGlove II is employed to capture the hand posture. This information is combined with the head and arm posture captured from the Microsoft Kinect. This is a new sensor solution for human-gesture capture. Based on the posture data from the CyberGlove II and Kinect, an effective and real-time human gesture recognition method is proposed. The gesture understanding approach based on an innovative combination of sensors is the main contribution of this paper. To verify the effectiveness of the proposed gesture

*Corresponding author: Institute for Media Innovation, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798. E-mail:drzhangzhijun@gmail.com

Figure 1: Human-robot social interaction. Human is on the right, and robot is on the left.

recognition method, a human body gesture dataset is built. The experimental results demonstrate that our approach can recognize the upper body gestures with high accuracy in real time. In addition, for robot motion generation and control, a novel online motion planning method is proposed. In order to generate appropriate dynamic motion, a quadratic programming (QP) based dual-arms kinematic motion generation scheme is proposed, and a simplified recurrent neural network is employed to solve the QP problem. The integration of handshake within the HRI system illustrates the effectiveness of the proposed online generation method.

1 Introduction

Recently, human-robot interaction (HRI) has drawn great attention in both academic and industrial communities. Being regarded as the sister community of human-computer interaction (HCI), HRI is still a relatively young field that began to emerge in the 1990s (Dautenhahn, 2007; Goodrich & Schultz, 2007). It is an interdisciplinary research field that requires contributions from mathematics, psychology, mechanical engineering, biology, computer science, etc. (Goodrich & Schultz, 2007)

HRI aims to understand and shape the interactions between humans and robots. Unlike the earlier human-machine interaction, more social dimensions must be considered in HRI, especially when interactive social robots are involved (Dautenhahn, 2007; Fong, Nourbakhsh, & Dautenhahn,

2003). In this case, robots should be believable. Moreover, humans prefer interacting with robots in the way they do with other people (Dautenhahn, 2007; Fong et al., 2003). Therefore, one way to increase believability would be to make the robot interact with human using the same modalities as human-human interaction. This includes verbal and body language as well as facial expressions. That is to say, the robots should be able to use these modalities for both perception and expression. Some social robots have already been proposed to achieve this goal. For instance, the Leonardo robot expresses itself using a combination of voice, facial and body expressions (Smith & Breazeal, 2007). Another example is the Nao humanoid robot[1] that can use vision along with gestures and body expression of emotions (Beck, Cañamero, et al., 2013). Different from these two robots, the Nadine robot is a highly realistic humanoid robot (Figure 1). This robot presents some different social challenges. In this paper, a human-robot interaction system that addresses some of these challenges is proposed. As shown in Figure 1, it supports one person to communicate and interact with one humanoid robot. In the proposed system, the human can naturally communicate with Nadine robot by using body language. The Nadine robot is able to express herself by using the combination of speech, body language and facial expressions. In this paper, the main research questions addressed are:

- How to establish the communication between human and robot by using body language;
- How to control a human-like robot so that it can sustain believable interaction with humans.

Verbal and non-verbal language are two main communication ways for human-human interaction. Verbal language has been employed in many HRI systems (Faber et al., 2009; Nickel & Stiefelhagen, 2007; Perzanowski, Schultz, Adams, Marsh, & Bugajska, 2001; Spiliotopoulos, Androutsopoulos, & Spyropoulos, 2001; Stiefelhagen et al., 2004, 2007). However, it still has some constraints. That is, speech recognition accuracy is likely to be affected by the background noise, human accents and device performance. Moreover, learning and interpreting the subtle rules of syntax and grammar in speech is also a difficult task. These factors limit the practical use of verbal language to some degree. On the other hand, non-verbal clues also convey rich communication message (Cassell et al., 2000; Mehrabian, 1971). Thus, one of our research

[1] http://www.aldebaran-robotics.com/

motivations is *to apply non-verbal language to human-robot social interaction.* More specifically, upper body gesture language is employed. Currently, 12 human upper body gestures are involved in the proposed system. These gestures are all natural ones with intuitive semantics. They are characterized by head, arm and hand posture simultaneously. It is worth noting that *human-object* interactions are involved in these gestures. *Human-object* interaction events manifest frequently during the human-human interaction in daily life. However, to our knowledge, they are largely ignored by the previous HRI systems.

The main challenge to apply upper body gesture language to human-robot interaction is how to enable Nadine robot to understand and react to human gestures accurately and in real time. To achieve this goal, two crucial issues need to be solved:

- First, appropriate human gesture-capture sensor solution is required. To recognize the 12 upper body gestures, head, arm and hand posture information are needed simultaneously. Because robustly obtaining hand posture based on the vision-based sensors (such as the RGB camera) is still a difficult task (Lu, Shark, Hall, & Zeshan, 2012; Teleb & Chang, 2012), the wearable CyberGlove II (Immersion, 2010) is employed. Using this device, high-accuracy hand posture data can be acquired stably. Meanwhile, the Microsoft Kinect (Shotton et al., 2011) is an effective and efficient low-cost depth sensor applied successfully to human body tracking. The skeleton joints can be extracted from the Kinect depth images (Shotton et al., 2011) in real time (30 fps). In our work, Kinect is applied to capturing the upper body (head and arm) posture information. Recently, Kinect 2 that supports tracking multiple people with better depth imaging quality has been released. Since our work investigates the HRI scenario that only involves one person, Kinect is sufficient to handle the human body tracking task;
- Secondly, an effective and real-time gesture recognition method should be developed. Based on the CyberGlove II and Kinect posture data, descriptive upper body gesture feature is proposed. To leverage the gesture understanding performance, LMNN distance metric learning method (Weinberger & Saul, 2009) is applied. Then, the energy-based LMNN classifier is used to recognize the gestures.

To evaluate the proposed gesture recognition method, a human upper body gesture dataset is

constructed. This dataset contains gesture samples from 25 people of different genders, body sizes and culture backgrounds. And, the experimental results demonstrate the effectiveness and efficiency of our method.

Overall, the main contributions of this paper include:

- A novel human gesture-capture sensor solution is proposed. That is, the CyberGlove II and Kinect are integrated to capture head, arm and hand posture simultaneously;
- An effective and real-time upper body gesture recognition approach is proposed;
- A novel online motion planning method is proposed for robot control;
- To support human to communicate and interact with robot using body language, a gesture understanding and human-robot interaction (GUHRI) system is built.

The remaining of this paper is organized as follows. The related work is discussed in Sec. 2. Sec. 3 gives an overview of the GURHI system. The human upper body gesture understanding method is illustrated in Sec. 4. The robot motion planning and control mechanism is described in Sec. 5. Sec. 6 introduces the scenario for human-robot interaction. Experiment and discussion are given in Sec. 7. Sec. 8 concludes the paper and discusses future research.

2 Related Work

HRI systems are constructed mainly based on verbal, non-verbal or multimodal communication modalities. As aforementioned in Section 1, verbal language still faces some constraints in practical applications. Our work focuses on studying how to apply non-verbal language to human-robot social interaction, especially using upper body gesture language. Some HRI systems have already employed body gesture language for human-robot communication. In (Waldherr, Romero, & Thrun, 2000), an arm gesture based interface for HRI was proposed. The user could control a mobile robot by using static or dynamic arm gestures. Hand gesture was used as the communication modality for HRI in (Brethes, Menezes, Lerasle, & Hayet, 2004). The HRI systems addressed in (Stiefelhagen et al., 2004, 2007) could recognize human's pointing gesture by using the 3D head and hand position information, and head orientation was further appended to

leverage the performance. In (Faber et al., 2009), the social robot could understand human's body language characterized by arm and head posture. Our proposition on non-verbal human-robot communication is different from the previous works mainly at two aspects. First, head, arm and hand posture are jointly captured to describe the 12 upper body gestures involved in the GUHRI system. Secondly, the gestures accompanied with human-object interaction can be understood by the robot. However, these gestures are always ignored by the previous HRI systems.

Body gesture recognition plays an important role in GUHRI system. According to the gesture-capture sensor type, gesture recognition systems can be categorized as encumbered and unencumbered ones (Berman & Stern, 2012). Encumbered systems require the users to wear physical assistive devices such as infrared responders, hand markers or data gloves. These systems are of high precision and fast response, and are robust to environment changes. Many encumbered systems have been proposed. For instance, two education systems (Adamo-Villani, Heisler, & Arns, 2007) were built for deaf by using data gloves and optical motion capture devices; Lu et al. (Lu et al., 2012) proposed an immersive virtual object manipulation system based on two data gloves and a hybrid ultrasonic tracking system. Although most commercialized gesture-capture devices are currently encumbered, unencumbered systems are expected as the future choice, especially the vision-based systems. With the emergence of low-cost 3D vision sensors, the application of such devices becomes a very hot topic in both research and commercial fields. One of the most famous examples is the Microsoft Kinect, it has been successfully employed in human body tracking (Shotton et al., 2011), activity analysis (Wang, Liu, Wu, & Yuan, 2012) and gesture understanding (Xiao, Yuan, & Thalmann, 2013). Even for other vision applications (such as scene categorization (Xiao, Wu, & Yuan, 2014) and image segmentation (Xiao, Cao, & Yuan, 2014)), Kinect also holds the potential to boost the performance. However, accurate and robust hand posture capture is still a difficult task for the vision-based sensors.

As discussed above, both encumbered and unencumbered sensors possess their intrinsic advantages and drawbacks. For the specific applications, they can be complementary. In the GUHRI system a tradeoff between the two kinds of sensors is made, that is the fine hand posture

is captured by the encumbered device (the CyberGlove II), while the rough upper body posture is handled by the unencumbered sensor (the Kinect).

The aim of the GUHRI system is to allow a user to interact with a humanoid robot named Nadine (Figure 1). In order to be believable, a humanoid robot should behave in a way consistent with its physical appearance (Beck, Stevens, Bard, & Cañamero, 2012). Therefore, due to its highly realistic appearance, the Nadine robot should rely on the same modalities as humans for communication. In other words, it should communicate using speech, facial expressions and body movements. This paper focuses mostly on body movements generation and describes a control mechanism that allows the robot to respond to the user's gestures in real time.

Humanoid social robots need to be able to display coordinated and independant arm movements depending on the actual situation. To do so, a kinematics model is necessary to generate motions dynamically. The dual-arms of humanoid robot is a redundant system, it is difficult to solve the inverse kinematics model directly. The classical approaches for solving the redundancy-resolution problem are the pseudoinverse based methods, i.e., one minimum-norm particular solution plus a homogeneous solution (Wang & Li, 2009). Specifically, at the joint-velocity level, the pseudoinverse-type solution can be formulated as

$$\dot{\theta}_{\mathrm{L}}(t) = J_{\mathrm{L}}^{\dagger}(\theta_{\mathrm{L}})\dot{r}_{\mathrm{L}} + (I_{\mathrm{L}} - J_{\mathrm{L}}^{\dagger}(\theta_{\mathrm{L}})J_{\mathrm{L}}(\theta_{\mathrm{L}}))w_{\mathrm{L}}, \tag{1}$$

$$\dot{\theta}_{\mathrm{R}}(t) = J_{\mathrm{R}}^{\dagger}(\theta_{\mathrm{R}})\dot{r}_{\mathrm{R}} + (I_{\mathrm{R}} - J_{\mathrm{R}}^{\dagger}(\theta_{\mathrm{R}})J_{\mathrm{R}}(\theta_{\mathrm{R}}))w_{\mathrm{R}}, \tag{2}$$

where θ_{L} and θ_{R} denote the joints of the left arm and the right arm respectively; $\dot{\theta}_{\mathrm{L}}$ and $\dot{\theta}_{\mathrm{R}}$ denote the joint velocities of the left arm and the right arm respectively; J_{L} and J_{R} are the Jacobian matrixes defined respectively as $J_{\mathrm{L}} = \partial f_{\mathrm{L}}(\theta_{\mathrm{L}})/\partial\theta_{\mathrm{L}}$ and $J_{\mathrm{R}} = \partial f_{\mathrm{R}}(\theta_{\mathrm{R}})/\partial\theta_{\mathrm{R}}$; $J_{\mathrm{L}}^{\dagger}(\theta_{\mathrm{L}}) \in R^{n\times m}$ and $J_{\mathrm{R}}^{\dagger}(\theta_{\mathrm{R}}) \in R^{n\times m}$ denote the pseudoinverse of the left arm Jacobian matrix $J_{\mathrm{L}}(\theta_{\mathrm{L}})$ and right arm Jacobian matrix $J_{\mathrm{R}}(\theta_{\mathrm{R}})$, respectively; $I_{\mathrm{L}} = I_{\mathrm{R}} \in R^{n}$ are the identity matrixes, and $w_{\mathrm{L}} \in R^{n}$ and $w_{\mathrm{R}} \in R^{n}$ are arbitrary vector usually selected by using some optimization criteria. The first terms of the right-hand of Equations 1 and 2 are the particular solutions (i.e., the minimum norm solutions), and the second terms are the homogeneous solutions. Pseudoinverse-based approaches need to compute matrix inverse, which may cost much time in real-time computation. In addition,

pseudoinverse-based approaches have an inner shortage, i.e., it cannot solve the inequality problems. In recent years, optimization methods are preferred (Cai & Zhang, 2012; Guo & Zhang, 2012; Kanoun, Lamiraux, & Wieber, 2011), and most of them focus on industrial manipulators and single robot (Z. Zhang & Zhang, 2012, 2013c).

The conventional redundancy resolution method is the pseudoinverse-type formulation, i.e., Equations 1 and 2. Based on such a pseudoinverse-type solution, many optimization performance criteria have been exploited in terms of manipulator configurations and interaction with the environment, such as joint-limits avoidance (Chan & Dubey, 1995; Ma & Watanabe, 2002), singularity avoidance (Taghirad & Nahon, 2008), and manipulability enhancement (Martins, Dias, & Alsina, 2006). Recent research shows that the solutions to redundancy resolution problems can be enhanced by using optimization techniques based on quadratic program (QP) methods (Cheng, Chen, & Sun, 1994). Compared with the conventional pseudoinverse-based solutions, such QP-based methods do not need to compute the inverse of the Jacobian matrix, and are readily to deal with the inequality and/or bound constraints. This is why QP-based methods have been employed. In (Cheng et al., 1994), considering the physical limits, Cheng et al. proposed a compact QP method to resolve the constrained kinematic redundancy problem. In (Z. Zhang & Zhang, 2013c), Zhang et al. implement a QP based two norm scheme on a planar six-DOF manipulator. However, the above methods only consider a single arm and are therefore not directly applicable for two arms of humanoid robot. This is why a QP-based dual-arms kinematic motion generation scheme is proposed, and a simplified recurrent neural network is employed to solve the QP problem.

3 System Overview

The proposed GUHRI system is able to capture and understand human upper body gestures and trigger the robot's reaction in real time accordingly. The GUHRI system is implemented using a framework called Integrated Integration Platform (I2P) that is specifically developed for integration. I2P was developed by the Institute for Media Innovation [2]. This framework allows for

[2] http://imi.ntu.edu.sg/Pages/Home.aspx

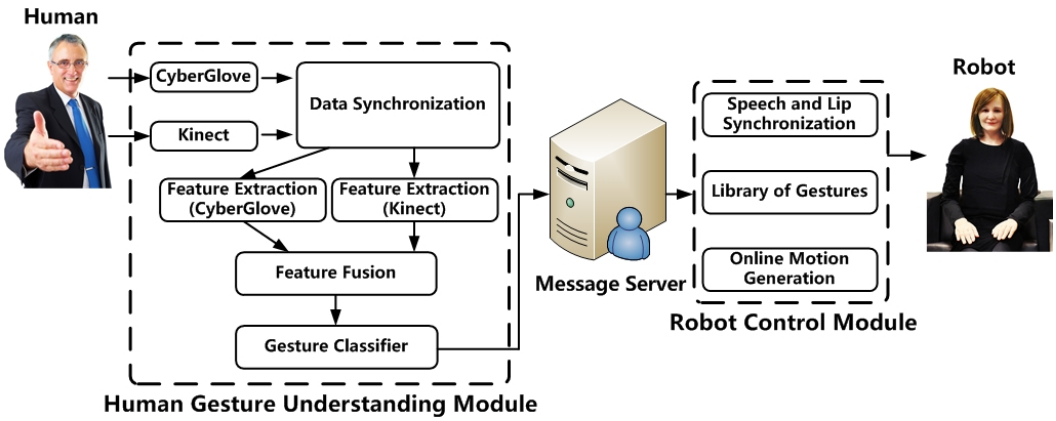

Figure 2: The GUHRI system architecture.

the link and integration of perception, decision and action modules within an unified and modular framework. The platform uses client-server communications between the different components. Each component has an I2P interface and the communication between the client and servers is implemented using thrift [3]. It should be noted that the framework is highly modular and components can be added to make the GUHRI system extendable. As shown in Fig. 2, the current GUHRI system is mainly composed of two modules. One is the *human gesture understanding module* that serves as the communication interface between human and robot, and the other is the *robot control module* proposed to control the robot's behaviors for interaction. At this stage, our system supports the interaction between one person and one robot.

One right hand CyberGlove II and one Microsoft Kinect are employed to capture human's hand and body posture information simultaneously for gesture understanding. This is a new gesture capturing sensor solution that is different from all the approaches introduced in Section 2. Specifically, CyberGlove is used to capture the hand posture, and Kinect is applied to acquiring the 3D position information of the human skeleton joints (including head, shoulder, limb and hand). At this stage, the GUHRI system relies on the upper body gestures triggered by the human's right hand and right arm.

Besides the CyberGlove, the user does not need to wear any other device. Thus, the proposed sensor solution does not exert heavy burden to make the user uncomfortable. Meanwhile, since

[3] http://thrift.apache.org/

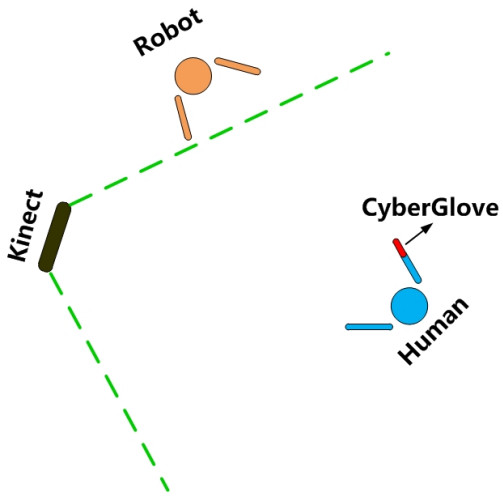

Figure 3: The GUHRI system deployment.

the CyberGlove II is involved in the system using Bluetooth, the user can move freely. In addition, GUHRI system is able to recognize gestures with human-object interaction, such as *"call"*, *"drink"*, *"read"* and *"write"* by fusing the hand and body posture information. These gestures are often ignored by the previous systems. However, these manifest frequently during the daily interaction between humans. These affect the interaction state abruptly and should be considered in HRI. Therefore, they should be regarded as the essential elements of the natural human-robot interaction. In our system, the robot is able to recognize and give meaningful responses to these gestures.

The first step of the gesture understanding phase is to synchronize the original data from the CyberGlove and Kinect. The descriptive features are then extracted from them respectively. The multimodal features are then fused to generate the unified input for the gesture classifier. Lastly, the gesture recognition and understanding results are sent to the robot control module via message server to trigger the robot's reaction.

The robot control module enables the robot to respond to the human's body gesture language. In our system, the robot's behavior is composed of three parts: body movement, facial expression and verbal language. Combining these modalities makes the robot more lifelike, and should enhance the users' interest during the interaction.

As shown in Figure 3, when the GUHRI system is running, the user wears the CyberGlove and

stands facing the robot for interaction. The Kinect is placed besides the robot to capture the user's body skeleton information.

4 Human Upper Body Gesture Understanding

As an essential part of the GUHRI system, the human upper body gesture understanding module plays an important role during the interaction. Its performance will highly affect the interaction experience. In this section, our upper body gesture understanding method by fusing the gesture information from CyberGlove and Kinect is illustrated in details. First, the body gestures included in the GUHRI system are introduced. The feature extraction pipelines for both CyberGlove and Kinect are then presented. To generate an integral gesture description, the multi-modal features from different sensors will be fused as the input for classifier. Aiming to enhance the gesture recognition accuracy, LMNN distance metric learning approach (Weinberger & Saul, 2009) is applied to mining the optimal distance measures. And, the energy-based classifier (Weinberger & Saul, 2009) is applied for decision making.

4.1 Gestures in the GUHRI System

At the current stage, 12 static upper body gestures are included in the GUHRI system. Since we only have one right hand CyberGlove, to obtain accurate hand posture information, all the gestures are mainly triggered by the human's right hand and right arm. The involved gestures can be partitioned into two categories, according to whether human-object interaction happens:

- **Category 1**: body gestures *without* human-object interaction;
- **Category 2**: body gestures *with* human-object interaction.

Category 1 contains 8 upper body gestures: *"be confident"*, *"have question"*, *"object"*, *"praise"*, *"stop"*, *"succeed"*, *"shake hand"* and *"weakly agree"*. Some gesture samples are shown in Figure 4. These gestures are natural and have intuitive meanings. They are related to the human's emotional state and behavior intention, not the ad hoc ones for specific applications. Therefore, gesture-to-meaning mapping is not needed in our system. Because the humans'

Figure 4: The **Category 1** upper body gestures. These gestures can be characterized by the body and hand posture information simultaneously.

behavior habits are not all the same, recognizing natural gestures is more challenging than the ad hoc ones. However, natural gestures are more meaningful for human-robot interaction. As exhibited in Figure 4, both hand and body posture information are required for recognizing these gestures. For instance, the upper body postures corresponding to *"have question"* and *"object"* are very similar. Without the hand posture, they are difficult to distinguish. The same thing also happens to *"have question"*, *"weakly agree"* and *"stop"*. That is, they correspond to similar hand gestures but very different upper body postures.

Category 2 is composed of 4 other pupper body gestures: *"call"*, *"drink"*, *"read"* and *"write"* (Figure 5). Being different from **Category 1** gestures, these 4 gestures are happening with human-object interactions. Existing system do not consider this kind of gestures (see Section 3). One main reason is that objects often causes the body occlusion, especially to the hand. In this case, vision-based hand gesture recognition methods are impaired. This is why the CyberGlove is

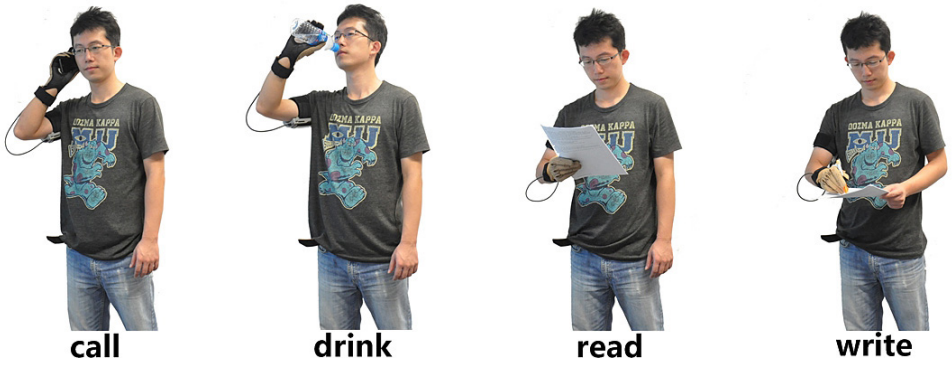

Figure 5: The **Category 2** upper body gestures. These gestures can be characterized by the body and hand posture information simultaneously.

employed to capture the hand posture. In the GUHRI system, **Category 2** gestures are recognized and affect the interaction in a realistic way. These gestures are also recognized based on the hand and upper body posture information.

As discussed above, the description of human's hand and upper body posture is the key to recognize and understand the 12 upper body gestures.

4.2 Feature Extraction and Fusion

In this subsection, we introduce the feature extraction methods for both human hand and upper body posture description. The multi-modal feature fusion approach is also illustrated.

4.2.1 Hand Posture Feature

The Immersion wireless CyberGlove II is employed as the hand posture capture device in the GUHRI system. As one of the most sophisticated and accurate data gloves, CyberGlove II provides 22 high-accuracy joint-angle measurements in real-time. These measurements reflect the bending degree of fingers and wrist. The 22 data joints (marked as big yellow or red dots) are located on the CyberGlove as shown in Figure 6. However, not all the joints are used. For the hand gestures in our application, we found that the wrist posture does not provide stable descriptive information. The wrist bending degrees of different people vary to large extent even

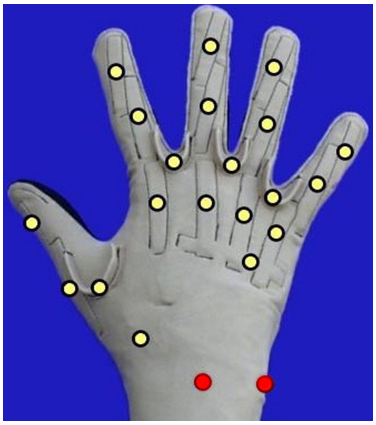

Figure 6: CyberGlove II data joints (Immersion, 2010).

for the same gesture. This phenomenon is related to different behaviour habits. This is why the two wrist data joints (marked as red) were discarded. A 20-dimensional feature vector F_{hand} is extracted from the 20 white data joints to describe the human hand posture as

$$F_{hand} = (h_1, h_2, h_3 \cdots h_{19}, h_{20}) \,, \tag{3}$$

where h_i is the bending degree corresponding to the white data joint i.

4.2.2 Upper Body Posture Feature

Using the Kinect sensor, we shape the human upper body posture intermediately using the 3D skeletal joint positions. For a full human subject, 20 body joint positions can be detected and tracked by the real-time skeleton tracker (Shotton et al., 2011) based on the Kinect depth frame. This is invariant to posture, body shape, clothing, etc. Each joint J_i is represented by 3 coordinates at the frame t as

$$J_i = (x_i(t), y_i(t), z_i(t)) \,. \tag{4}$$

However, not all the 20 joints are necessary for upper body gesture recognition. As aforementioned, head and right arm are highly correlated with the 12 upper body gestures (Figure 4 and Figure 5). For efficiency, only 4 upper body joints are chosen as the descriptive joints for gesture understanding. These are *"head"*, *"right shoulder"*, *"right elbow"* and *"right hand"* that are shown as the green dots in Figure 7.

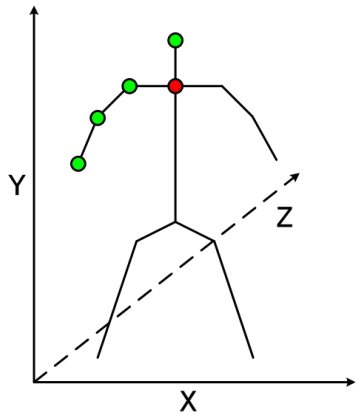

Figure 7: The selected body skeletal joints.

Directly using the original 3D joint information for body posture description is not stable, because it is sensitive to the relative position between human and Kinect. Solving this problem by restricting the human's position is not appropriate for interaction. In (Wang et al., 2012), human action is recognized by using the pairwise relative positions between all joints, which is robust to human-Kinect relative position. Inspired by this work, a simplified solution is proposed. First, the *"middle of the two shoulders"* joint (red dot in Figure 7) is selected as the reference joint. The pairwise relative positions between the 4 descriptive joints and the reference joint are then computed for body posture description as

$$J_{sr} = J_s - J_r \,, \tag{5}$$

where J_s is the descriptive joint and J_r is the reference joint. With this processing, J_{sr} is less sensitive to the human-Kinect relative position. It is mainly determined by the body posture. The *"middle of the two shoulders"* was chosen as the reference joint because it can be robustly detected and tracked in most cases. Moreover, it is rarely occluded by the limbs or the objects when the gestures in GUHRI system happen. Finally, an upper body posture feature vector F_{body} of 12 dimensions is constructed by combining the 4 pairwise relative positions as

$$F_{body} = (J_{1r}, J_{2r}, J_{3r}, J_{4r}) \,, \tag{6}$$

where J_{1r}, J_{2r}, J_{3r} and J_{4r} are the pairwise relative positions.

4.2.3 Feature Fusion

From the CyberGlove II and Kinect, two multimodal feature vectors: F_{hand} and F_{body} are extracted to describe the hand posture and upper body posture respectively. To fully understand the upper body gestures, the joint information of the two feature vectors is required. Both of them are essential for the recognition task. However, the two feature vectors locate in different value ranges. Simply combining them as the input for classifier will yield performance bias on the feature vector of low values. To overcome this difficulty, we scale them into similar ranges before feature fusion. Supposing F_i is one dimension of F_{hand} or F_{body}, F_i^{max} and F_i^{min} are the corresponding maximum and minimum value in the training set. Then F_i can be normalized as

$$\hat{F}_i = \frac{F_i - F_i^{min}}{F_i^{max} - F_i^{min}}\,, \tag{7}$$

for both training and test.

After normalization, the effectiveness of the two feature vectors for gesture recognition will be balanced. Finally, they are fused to generate an integral feature vector by concatenation as

$$\vec{F} = (\hat{F}_{hand}, \hat{F}_{body})\,. \tag{8}$$

This process results in an 32-dimensional feature vector $\vec{F}$ used for upper body gesture recognition.

4.3 Classification Method

Using $\vec{F}$ as the input feature, the upper body gestures will be recognized by template matching based on the *energy-based LMNN classifier* proposed in (Weinberger & Saul, 2009)[4]. It is derived from the energy-based model (Chopra, Hadsell, & LeCun, 2005) and the LMNN distance metric learning method (Weinberger & Saul, 2009). The latter part is the key to constructing this classifier. LMNN distance metric learning approach is proposed to seek the best distance measure

[4]The source code is available at http://www.cse.wustl.edu/~kilian/code/lmnn/lmnn.html.

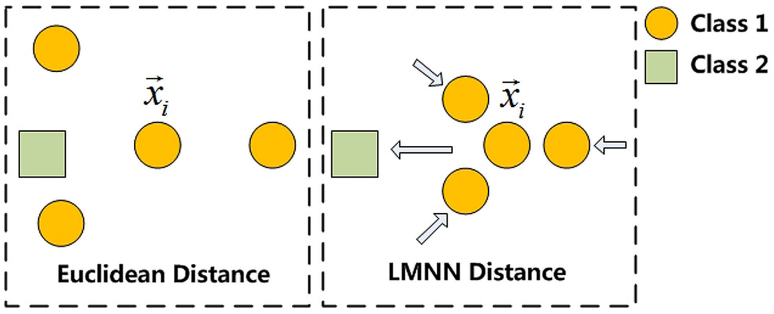

Figure 8: Illustration of the LMNN distance metric learning.

for the *k*-nearest neighbor (KNN) classification rule (Cover & Hart, 1967). As one of the oldest methods for pattern recognition, KNN classifier is very simple to implement and use. Nevertheless, it can still yield comparative results in certain domains such as object recognition and shape matching (Belongies, Malik, & Puzicha, 2002). And it also has been applied to action recognition (Müller & Röder, 2006).

The KNN rule classifies each testing sample by the majority label voting among its *k*-nearest training samples. Its performance crucially depends on how to compute the distances between different samples for the *k* nearest neighbors search. Euclidean distance is the most widely used distance measure. However, it ignores any statistical regularities that may be estimated from the training set. Ideally, the distance measure should be adjusted according to the specific task being solved. To achieve better classification performance, LMNN distance metric learning method is proposed to mine the best distance measure for the KNN classification.

Let $\{(\vec{x}_i, y_i)\}_{i=n}^{n}$ be a training set of *n* labeled samples with inputs $\vec{x}_i \in \mathbb{R}^d$ and class labels y_i. The main goal of LMNN distance metric learning is to learn a linear transformation $\mathbf{L} : \mathbb{R}^d \to \mathbb{R}^d$ that is used to compute the square sample distances as

$$\mathcal{D}(\vec{x}_i, \vec{x}_j) = \|\mathbf{L}(\vec{x}_i - \vec{x}_j)\|^2 . \tag{9}$$

Using $\mathcal{D}(\vec{x}_i, \vec{x}_j)$ as the distance measure tends to optimize the KNN classification by making each input $\vec{x}_i$ have *k* nearest neighbors that share the same class label y_i to the greatest possibility. Figure 8 gives an intuitive illustration on LMNN distance metric learning. Compared with Euclidean distance, LMNN distance tries to pull the nearest neighbors of class y_i closer to $\vec{x}_i$,

meanwhile push the neighbors from different classes away. Under the assumption that the training set and the test set keep the similar feature distribution, LMNN distance metric learning can help to improve the KNN classification result.

The energy-based LMNN classifier makes use of both the $\mathcal{D}(\vec{x}_i, \vec{x}_j)$ distance measure and the loss function defined for LMNN distance metric learning. It constructs an energy-based criterion function, and the testing sample is assigned to the class which yields the minimum loss value. Because the related theory is sophisticated, we do not give the detailed definition on the energy-based LMNN classifier here. The readers can turn to (Weinberger & Saul, 2009) for reference.

The next section describes the robot control mechanism that make it react to human gestures.

5 Robot Motion Planning and Control

The Nadine robot (Figure 1) is a realistic human-size robot developed by Kokoro Company, Ltd.[5]. It has 27 degrees of freedom and uses pneumatic motors to display natural looking movements. Motion planning and control are always an important issue for robots (Miyashita & Ishiguro, 2004) and are becoming a necessary and promising research area (Pierris & Lagoudakis, 2009; Takahashi, Kimura, Maeda, & Nakamura, 2012). They allow the synchronization of animations (pre-defined and online animations), speech and gaze. The following sections describe the core functionalities of the Nadine robot controller. This includes lip synchronisation, synchronization of pre-defined gestures and facial expressions and online motion generation.

5.1 Lip synchronisation

Lip synchronisation is part of the core function of the Nadine robot controller. It ensures that the Nadine robot looks natural when talking. However, implementing this on a robot such as the Nadine is a challenging task. In one hand, the Nadine robot is physically realistic raising users' expectations, on the other hand, the robot has strong limitations in terms of the range and speed of

[5] http://www.kokoro-dreams.co.jp/english/

Figure 9: Lip-synch for part of the sentence *“I am Nadine”*.

movements that it can achieve. The Cerevoice text-to-speech library[6] is used to extract the phonemes as well as to synthesise the speech. Figure 9 illustrates the process for the beginning of the sentence “I am Nadine”. First, the following phonemes are extracted: “sil”, “ay”, “ax”, “m”, “n”, “ey”,“d”, “iy”, “n” along with their durations. Due to the Nadine robot’s velocity limits, it is not possible to generate lips movements for all the phonemes. This is why, to maintain the synchronisation any phonemes that last less than 0.1 second is ignored and the duration of the next one is extended by the same amount. In the Figure 9 example “ax” is removed and “m” is extended, “n” is removed and “ey” is extended, and “d” is removed and “iy” is extended. The phonemes are then mapped to visemes that were designed by a professional animator. Figure 9 shows examples of two visemes (Frames 3 and 10). The transitions between phonemes is done using cosine interpolation (see Figure 9 frames 4 to 9). Moreover, the robot cannot display a “O” mouth movement along with a “Smile”. Therefore, if a forbidden transition is needed, a closing mouth movement is generated prior to display the next viseme. The synchronisation is done so that the pre-defined viseme position is reached at the end of each phoneme.

5.2 Library of Gestures and Idle Behaviours

In addition to the lip-synch animation generator, a professional animator is designing pre-defined animations for the Nadine Robot. This is used to display iconic gestures such as waving hello. The pre-defined gestures also include facial and bodily emotional expressions (Figure 10).

[6] http://www.cereproc.com/en/products/sdk

Figure 10: Examples of body movements and facial expressions from the library of gestures.

Providing they do not require the same joints, they are dynamically combined to create richer display. We are also designing a generator for idle behaviours to avoid having the robot remaining completely static as this would look unnatural. To do so, we plan to use the well established Perlin Noise to generate movements as this can be modulated to express emotions (Beck, Hiolle, & Cañamero, 2013).

5.3 Online Motion Generation

The kinematics model of Nadine robot is a dual-arms kinematic model. Kinematics model of Nadine robot includes two parts, i.e., forward kinematics model and inverse kinematic model. Forward kinematics model outputs the end-effector (hand) trajectories of robot if the joint vector of dual-arms is given, and inverse kinematic model outputs joint vector of dual-arms if the end-effector (hand) path is known. Mathematically, given joint-space vector $\theta(t) \in R^n$, the end-effector position/orientation vector $r(t) \in R^m$ can be formulated as the following forward kinematic equation:

$$r(t) = f(\theta(t)), \tag{10}$$

where $f(\cdot)$ is a smooth nonlinear function, which can be obtained if the structure and parameters of a robot is known; n is the dimension of joint space; m is the dimension of end-effector

Cartesian space. Conversely, given end-effector position/orientation vector $r(t) \in R^m$, joint-space vector $\theta(t) \in R^n$ can be denoted by

$$\theta(t) = f^{-1}(r(t)), \tag{11}$$

where $f^{-1}(\cdot)$ is the inverse function of $f(\cdot)$ in Equation 10. For a redundant arm system, i.e., $n > m$, the difficulty is that inverse kinematics Equation 11 is usually nonlinear and under-determined, and is difficult (even impossible) to solve. The key of online motion generation is how to solve the inverse kinematics problem.

In our setting (Figure 11), the robot is expected to generate social gestures and motions dynamically according to the situation. For instance, handshake is commonly used as a greeting at the beginning and end of an interaction. Moreover, this allows the robot to communicate through touch which is common in human-human interaction. This kind of gestures cannot be included in the pre-defined library as they need to be adapted to the current user's position on-the-fly. In order to generate motion dynamically, the forward kinematic equations of dual-arms are first built. Then, they are integrated into a quadratic programming formulation. After that, we use a simplified recurrent neural network to solve such a quadratic programming. Specifically, when the robot recognizes that the user wants to shake hands with her, the robot stretches out her hand to the user and shake hand with the user.

The forward kinematics model considers the robot's arms. Each arm has 7 degrees-of-freedom. Given the left arm end-effector position vector $p_{\text{endL}} \in R^m$, the right arm end-effector position vector $p_{\text{endR}} \in R^m$ and their corresponding homogeneous representation $r_L = r_R \in R^{m+1}$ with superscript $^{\text{T}}$ denoting the transpose of a vector or a matrix, we can obtain the homogeneous representations r_{L} and r_{R} from the following chain formulas, respectively.

$$r_L(t) = f_{\text{L}}(\theta_{\text{L}}) = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \cdot {}^3_4T \cdot {}^4_5T \cdot {}^5_6T \cdot {}^6_7T \cdot p_{\text{endL}}, \tag{12}$$

$$r_R(t) = f_{\text{R}}(\theta_{\text{R}}) = {}^0_8T \cdot {}^8_9T \cdot {}^9_{10}T \cdot {}^{10}_{11}T \cdot {}^{11}_{12}T \cdot {}^{12}_{13}T \cdot {}^{13}_{14}T \cdot p_{\text{endR}}, \tag{13}$$

where ${}^i_{i+1}T$ with $i = 0, 1, \cdots, 14$ denote the homogeneous transform matrixes. In this paper, $n = 7$ and $m = 3$.

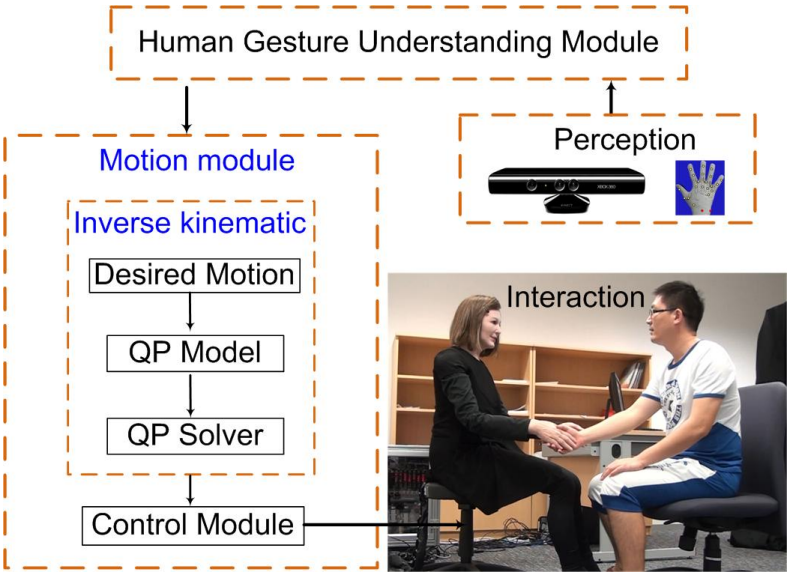

Figure 11: System chart of on line motion generation with handshake as an example.

Inspired by the work on one-arm redundant system (Z. Zhang & Zhang, 2013c), we try to build a model based on quadratic programming as shown below.

$$\text{minimize} \qquad \dot{\vartheta}^{\mathrm{T}}(t) M \dot{\vartheta}(t)/2 \tag{14}$$

$$\text{subject to} \qquad \jmath(\vartheta)\dot{\vartheta}(t) = \dot{\Upsilon}(t), \tag{15}$$

$$\vartheta^{-}(t) \leq \vartheta(t) \leq \vartheta^{+}(t), \tag{16}$$

$$\dot{\vartheta}^{-}(t) \leq \dot{\vartheta}(t) \leq \dot{\vartheta}^{+}(t), \tag{17}$$

where $\vartheta(t) = [\theta_{\mathrm{L}}^{\mathrm{T}}, \theta_{\mathrm{R}}^{\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$; $\vartheta^{-}(t) = [\theta_{\mathrm{L}}^{-\mathrm{T}}, \theta_{\mathrm{R}}^{-\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$; $\vartheta^{+}(t) = [\theta_{\mathrm{L}}^{+\mathrm{T}}, \theta_{\mathrm{R}}^{+\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$; $\dot{\vartheta}(t) = d\vartheta/dt = [\dot{\theta}_{\mathrm{L}}^{\mathrm{T}}, \dot{\theta}_{\mathrm{R}}^{\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$; $\dot{\vartheta}^{-}(t) = [\dot{\theta}_{\mathrm{L}}^{-\mathrm{T}}, \dot{\theta}_{\mathrm{R}}^{-\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$; $\dot{\vartheta}^{+}(t) = [\dot{\theta}_{\mathrm{L}}^{+\mathrm{T}}, \dot{\theta}_{\mathrm{R}}^{+\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$. $\dot{\Upsilon}(t) = [\dot{r}_{\mathrm{L}}^{\mathrm{T}}; \dot{r}_{\mathrm{R}}^{\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$. Matrix $\jmath$ is composed by Jacobian matrixes J_{L} and J_{R}; M is a $n \times n$ identity matrix. Specifically, $M \in R^{2n \times 2n}$ is an identity matrix, and

$$\jmath = \begin{bmatrix} J_{\mathrm{L}} & \mathbf{0}_{m \times n} \\ \mathbf{0}_{m \times n} & J_{\mathrm{R}} \end{bmatrix} \in R^{2m \times 2n}.$$

For the sake of calculations, the QP-based coordinated dual-arm scheme can be formulated as the

following expression constrained by an equality and inequality.

$$\text{minimize} \qquad \|\dot{\vartheta}(t)\|^2/2 \tag{18}$$

$$\text{subject to} \qquad J(\vartheta)\dot{\vartheta}(t) = \dot{\Upsilon}(t), \tag{19}$$

$$\xi^-(t) \leqslant \dot{\vartheta}(t) \leqslant \xi^+(t), \tag{20}$$

where $\|\cdot\|$ denotes the two norm of a vector or a matrix. Equation 18 is the simplification of Equation 14. Equation 20 is transformed by Equations 16 and 17. In Equation 20, the ith components of $\xi^-(t)$ and $\xi^+(t)$ are $\xi_i^-(t) = \max\{\dot{\vartheta}_i^-, \nu(\vartheta_i^-(t) - \vartheta_i)\}$ and $\xi_i^+(t) = \min\{\dot{\vartheta}_i^+, \nu(\vartheta_i^+(t) - \vartheta_i)\}$ with $\nu = 2$ being used to scale the feasible region of $\dot{\vartheta}$. $\vartheta^- = [\vartheta_{\mathrm{L}}^{-\mathrm{T}}, \vartheta_{\mathrm{R}}^{-\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$; $\vartheta^+ = [\vartheta_{\mathrm{L}}^{+\mathrm{T}}, \vartheta_{\mathrm{R}}^{+\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$. $\dot{\vartheta}^- = [\dot{\vartheta}_{\mathrm{L}}^{-\mathrm{T}}, \dot{\vartheta}_{\mathrm{R}}^{-\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$; $\dot{\vartheta}^+ = [\dot{\vartheta}_{\mathrm{L}}^{+\mathrm{T}}, \dot{\vartheta}_{\mathrm{R}}^{+\mathrm{T}}]^{\mathrm{T}} \in R^{2n}$. In the subsequent experiments, the physical limits

$\vartheta_{\mathrm{L}}^+ = [\pi/20, \pi/10, \pi/8, \pi/2, 0, 2\pi/3, \pi/2]^{\mathrm{T}}$,

$\vartheta_{\mathrm{L}}^- = [0, -3\pi/10, -7\pi/120, 0, -131\pi/180, 0, \pi/9]^{\mathrm{T}}$,

$\vartheta_{\mathrm{R}}^+ = [0, 3\pi/10, 7\pi/120, \pi, 0, 2\pi/3, -\pi/2]^{\mathrm{T}}$, and

$\vartheta_{\mathrm{R}}^- = [-\pi/20, -\pi/10, -\pi/8, \pi/2, -131\pi/180, \quad 0, -8\pi/9]^{\mathrm{T}}$.

Firstly, according to (Z. Zhang & Zhang, 2013c), Equations 18-20 can be converted to a linear variational inequality. That is to find a solution vector $u^* \in \Omega$ w.r.t.

$$(u - u^*)^{\mathrm{T}}(\Gamma u^* + q) \geq 0. \; \forall u \in \Omega \tag{21}$$

Secondly, Equation 21 is equivalent to the following system of piecewise-linear equations (Z. Zhang & Zhang, 2013c):

$$\Phi_\Omega(u - (\Gamma u + q)) - u = 0, \tag{22}$$

where $\Phi_\Omega(\cdot) : \; R^{2n+2m} \rightarrow \Omega$ is a projection operator, i.e.,

$$\begin{cases} u_i^-, & \text{if } u_i < u_i^-, \\ u_i, & \text{if } u_i^- \leqslant u_i \leqslant u_i^+, \; \forall i \in \{1, 2, \cdots, n+m\}. \\ u_i^+, & \text{if } u_i > u_i^+, \end{cases}$$

In addition, $\Omega = \{u \in R^{2n+2m} | u^- \leq u \leq u^+\} \subset R^{2n+2m}$; $u \in R^m$ is the primal-dual decision vector; $u^- \in R^m$ and $u^+ \in R^m$ are the lower and upper bounds of u, respectively; ω is usually set

a sufficiently large value (e.g., in the simulations and experiments afterward, $\varpi := 10^{10}$). Specifically,

$$u = \begin{bmatrix} \vartheta(t) \\ \iota \end{bmatrix} \in R^{2n+2m},\ u^{+} = \begin{bmatrix} \zeta^{+}(t) \\ \omega 1_{\iota} \end{bmatrix} \in R^{2n+2m}, u^{-} = \begin{bmatrix} \zeta^{-}(t) \\ -\omega 1_{\iota} \end{bmatrix} \in R^{2n+2m},$$

$$\Gamma = \begin{bmatrix} M & -\jmath^{\mathrm{T}} \\ \jmath & 0 \end{bmatrix} \in R^{(2n+2m)\times(2n+2m)}, q = \begin{bmatrix} 0 \\ -\dot{\Upsilon} \end{bmatrix} \in R^{n+m},\ 1_{\mathrm{v}} := [1, \cdots, 1]^{\mathrm{T}}.$$

Thirdly, being guided by dynamic-system-solver design experience (Y. Zhang, Tan, Yang, Lv, & Chen, 2008; Y. Zhang, Wu, Zhang, Xiao, & Guo, 2013; Z. Zhang & Zhang, 2013a, 2013b), we can adopt the following neural dynamics (the simplified recurrent neural network (Y. Zhang et al., 2008)) to solve Equation 22.

$$\dot{u} = \gamma P_{\Omega}(u - (Mu + q)) - u, \tag{23}$$

where γ is a positive design parameter used to scale the convergence rate of the neural network. The lemma proposed in (Y. Zhang et al., 2008) guarantees the convergence of neural network formulated by Equation 23 (with proof omitted due to space limitation).

Lemma: Assume that the optimal solution $\vartheta^{\star}$ to the strictly-convex QP problem formulated by Equations 18-20 exists. Being the first $2n$ elements of state $u(t)$, output $\vartheta(t)$ of the simplified recurrent neural network in Equation 23 is globally exponentially convergent to $\vartheta^{\star}$. In addition, the exponential-convergence rate is proportional to the product of γ and the minimum eigenvalue of M (Y. Zhang et al., 2008).

6 Human-Robot Interaction

As a case study for the GUHRI system, a scenario was defined in which a user and the robot interact in a classroom. The robot is the lecturer, and the user is the student. The robot is a female named “Nadine”. Nadine can understand the 12 upper body gestures described in Section 4.1 and react to the users’ gestures accordingly. In our system, Nadine is human-like and capable of reacting by combining body movement, facial expression and verbal language (see Section 5). In

Table 1: The scenario for human-robot interaction.

Human gestures	Nadine's response	
	Non Verbal	Verbal
"be confident"	happy	It is great to see you so confident.
"have question"	moderate	What is your question?
"object"	sad	Why do you disagree?
"praise"	happy	Thank you for your praise.
"stop"	moderate	Why do you stop me?
"succeed"	happy	Well done. You are successful.
"shake hand"	happy	Nice to meet you.
"weakly agree"	head nod	OK, we finally reach an agreement.
"call"	head shake	Please turn off your phone.
"drink"	moderate	You can have a drink. No problem.
"read"	moderate	Please, take your time and read it carefully.
"write"	moderate	If you need time for taking notes, I can slow my presentation.

this way, Nadine's reactions provide the user with vivid feedback. Figure 10 shows some examples of Nadine's body movements along with corresponding facial expressions. Nonverbal behaviors can help to structure the processing of verbal information as well as giving affective feedback during the interaction (Cañamero & Fredslund, 2001; Krämer, Tietz, & Bente, 2003). Thus, body movements and facial expressions are expected to enhance the quality of the interaction with Nadine.

In this scenario, Nadine's behaviors are triggered by the users' body language. Her reactions are consistent with the defined scenario (see Table 1). It should be noted that because it is difficult to fully describe the robot's body actions, the robot's movements and emotional display are described at a high level. All the 12 upper body gestures are involved. The GHURI system can also handle unexpected situations during the interaction. For example, Nadine can react appropriately even if the user suddenly answers a coming phone call.

7 Experiment and Discussion

The experiment comprises two main parts. First, our upper body gesture recognition method is tested. Then, the effectiveness of the proposed online motion generation approach is verified by executing handshake between the user and the robot.

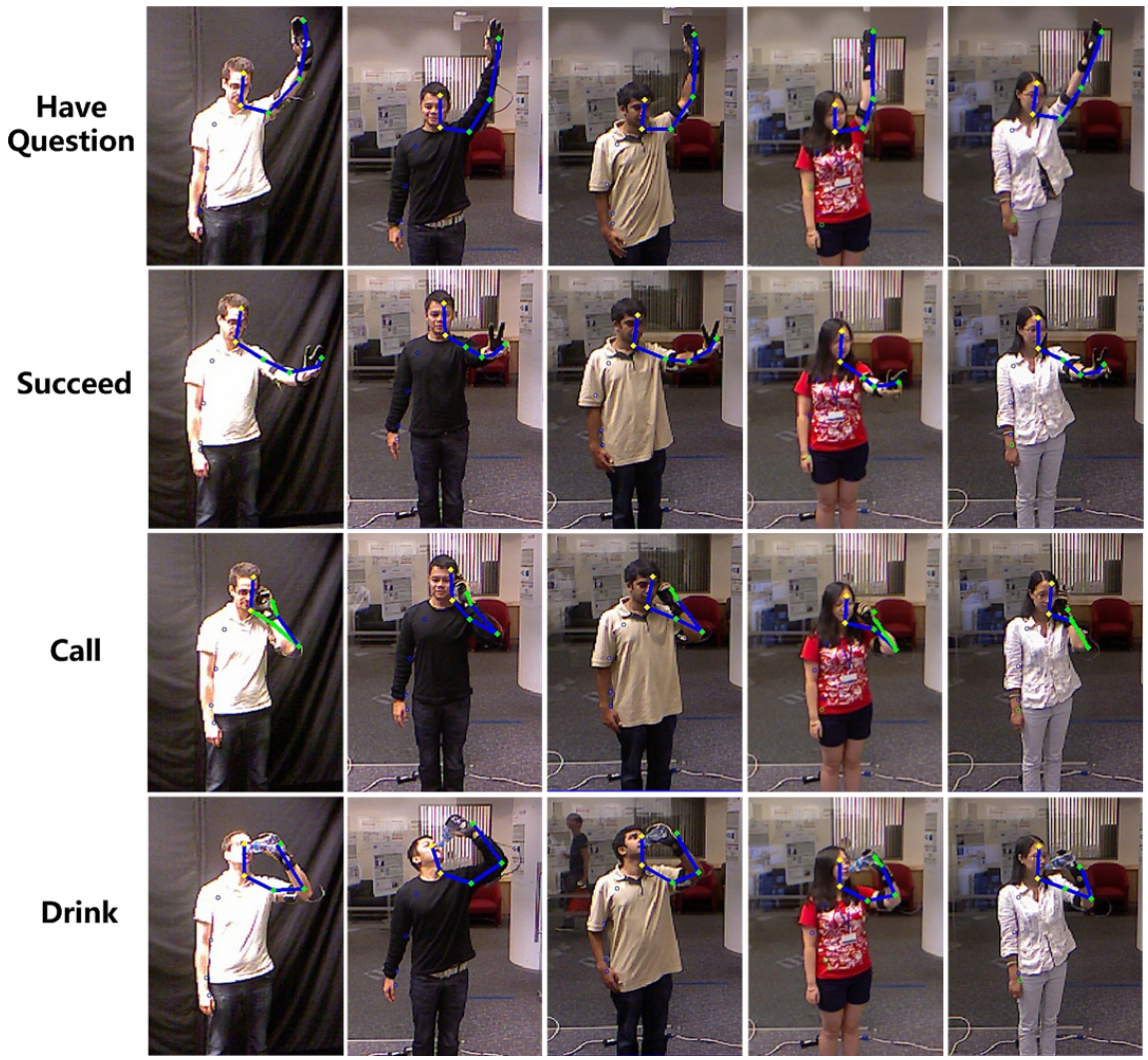

Figure 12: Some gesture samples captured from different volunteers. These people are of different genders, body sizes and races. They executed the gestures according to their own habits.

7.1 Upper Body Gesture Recognition Test

A human upper body gesture dataset was built to test the proposed gesture recognition method. This dataset involves all the 12 upper body gestures mentioned in Section 4.1. The samples are captured from 25 volunteers of different genders, body sizes and races. During the sample collection, no strict constraint was imposed to the people. They carried out the gestures according to their own habits. The user-Kinect relative position was also not strictly limited. For convenience, the CyberGlove II was pre-calibrated for all the people with a standard calibration.

Due to the dataset collection setup, large diversities may exist among the gesture samples from different people. This will yield challenges on body gesture recognition. Figure 12 exhibits parts of the **Category 1** and **Category 2** gesture samples (*"have question"*, *"succeed"*, *"call"* and *"drink"*) captured from 5 people for comparison. For the sake of brevity, not all the gestures are shown. The 5 descriptive and reference skeletal joints proposed in Section 4.2.2 are marked as the color dots in Figure 12. And, they are connected by the straight segments to shape the upper body posture intuitively. From the exhibited samples, we can observe that:

- For the different people, the listed body gestures can indeed be differentiated from the hand and upper body posture information. And, the people execute the gestures differently to some degree. As aforementioned, this phenomenon leads to challenges on upper body gesture recognition;
- For different people and gestures, the 5 skeletal joints employed for gesture recognition can be tracked robustly, even when human-object interaction occurs. Generally, their resulting positions are accurate for gesture recognition. Meanwhile, the CyberGlove II is a human-touch device that can capture the hand posture robustly to yield high-accuracy data. Thus, the proposed human gesture-capture sensor solution can stably acquire available data for gesture recognition.

For each gesture, one key snapshot is picked up to built the dataset among all the 25 people. As a consequence, the resulting dataset contains $25 \times 12 = 300$ gesture samples in all. During experiment, the samples are randomly split into the training and testing set for 5 times, and the average classification accuracy and standard deviation are reported.

The KNN classifier is employed as the baseline to make comparison with the energy-based LMNN classifier. They are compared both on the items of classification accuracy and time consumption. The KNN classifier will run with different kinds of distance measures. Following the experiment setup in (Weinberger & Saul, 2009), *"k"* is set as 3 in all cases. Because the training sample number is a crucial factor that affects the classification accuracy, the results of two classifiers are compared corresponding to different amounts of training samples. For each class, the training sample number will increase from 4 to 14 with the step size 2.

Table 2: Classification result (%) of the constructed upper body gesture dataset. The best performance is shown in boldface. Standard deviations are in parentheses.

Classifiers	Training sample number per class		
	4	6	8
KNN (Euclidean)	86.51(±2.89)	89.56(±2.43)	91.47(±2.21)
KNN (PCA)	73.81(±4.78)	84.04(±5.41)	79.31(±3.09)
KNN (LDA)	79.68(±5.33)	90.44(±2.56)	92.35(±2.44)
KNN (LMNN)	86.67(±2.18)	90.35(±2.56)	92.16(±1.80)
Energy (LMNN)	**90.00**(±**3.40**)	**92.28**(±**0.85**)	**94.31**(±**2.04**)
	10	12	14
KNN (Euclidean)	93.00(±1.15)	93.33(±0.73)	92.27(±2.30)
KNN (PCA)	86.11(±2.75)	88.21(±4.17)	86.67(±3.70)
KNN (LDA)	92.44(±1.34)	94.74(±0.84)	93.48(±1.38)
KNN (LMNN)	93.67(±1.34)	93.85(±1.48)	93.48(±1.15)
Energy (LMNN)	**95.22**(±**1.50**)	**95.64**(±**1.39**)	**96.52**(±**2.37**)

Other two well known distance metric learning methods: PCA (Jolliffe, 1986) and LDA (Fisher, 1936) are employed for comparison with the LMNN distance metric learning approach. For PCA, the first 10 eigenvectors are used to capture roughly 90% of the sum of eigenvalues. And, the first 6 eigenvectors are employed for LDA. The distance measures yielded by PCA and LDA will be applied to the KNN classifier.

Table 2 lists the classification results yielded by the different classifier and distance measure combinations. It can be observed that:

- The 12 upper body gestures in the dataset can be well recognized by the proposed gesture recognition method. More than 95.00% classification accuracy can be achieved if enough training samples are used. With the increase of training sample amount, the performance is generally enhanced consistently;
- Corresponding to all the training sample numbers, the energy-based LMNN classifier can yield the highest classification accuracy. Even with small number (such as 4) of training samples, it can still achieve relative good performance (90.00%). When the training sample number reaches 14, the classification accuracy (96.52%) is nearly satisfied for practical use. And, its standard deviations are relatively low in most cases, which means that the

Table 3: Average testing time consumption (ms) per sample. The program is running on the computer with Intel (R) Core (TM) i5-2430M @ 2.4GHz (only using one core).

Classifiers	Training sample number per class		
	4	6	8
KNN (LDA)	0.0317	0.0398	0.0414
KNN (LMNN)	0.0239	0.0282	0.0328
Energy (LMNN)	0.0959	0.1074	0.1273
	10	12	14
KNN (LDA)	0.0469	0.0498	0.0649
KNN (LMNN)	0.0342	0.0418	0.0525
Energy (LMNN)	0.1359	0.1610	0.1943

energy-based LMNN classifier is robust to the gesture diversities among people;

- KNN classifier can also yield good result on this dataset. However, it is inferior to the energy-based LMNN classifier. Compared to Euclidean distance, LMNN distance metric learning method can improve the performance of KNN classifier consistently in most cases. However, it works much better on the energy-based model;
- PCA does not work well on this dataset. Its performance is even worse than the basic Euclidean distance. The reason may be that PCA needs large number of training samples to obtain the satisfied distance measures (Osborne & Costello, 2004). That is the limitation for the practical applications.
- LDA also achieves good performance for the upper body gesture recognition. However, it is still consistently inferior to energy-based LMNN, especially when the training sample number is small. For example, when the training sample number is only 4, energy-based LMNN's accuracy (90.00%) is significantly better than that of LDA (79.68%) by a large margin (10.32%).

Besides the classification accuracy, the testing time consumption is also what we concern about. The reason is that the GUHRI system should run in real time for good HRI experience. According to the classification results in Table 2, the energy-based LMNN classifier, LMNN KNN classifier and LDA KNN classifier are the threep strongest ones for gesture recognition. Here, the comparison on their testing time is also made. Table 3 lists the average running time per testing

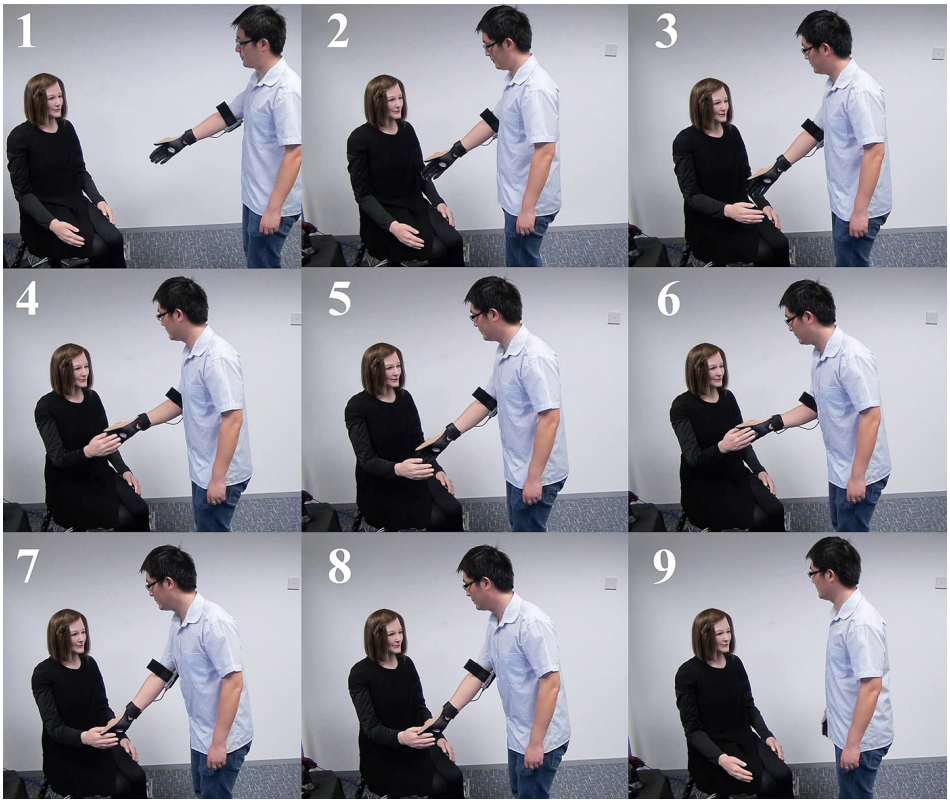

Figure 13: Snapshots of shaking hand between human and Nadine robot.

sample of the three classifiers, corresponding to different amounts of training samples. We can see that both of the three classifiers are extremely fast under our experimental conditions. And, the time consumption mainly depends on the number of training samples. Frankly, the LDA KNN classifier and LMNN KNN classifier are much faster than the energy-based LMNN classifier. If huge number of training samples are used (such as tens of thousands), the LDA KNN classifier and LMNN KNN classifier will be the better choice to achieve the balance between classification accuracy and computational efficiency.

7.2 The Motion Generation and Control Effectiveness

In this section, the effectiveness of the motion generation (Section 5.3) and control are illustrated within the case study described in Section 6. At the beginning of the interaction, a user can shake hand with Nadine robot. The snapshots of the handshake are shown in Figure 13. From the figure,

we can see that, when the user is far from the Nadine robot, Nadine robot knows that it is too far to shake hands, and she will ask the user to come closer. When the user come closer and give a handshake gesture, Nadine robot will recognize it, and stretch out her hand to shake hands with the user whilst saying "nice to meet you". After the handshake, she puts back her hand to the initial position and is ready for the next movement. From Figure 13, we can see the Nadine robot is able to shake hands with people well. This situation demonstrates that the proposed QP based on-line motion generation approach is effective, applicable and well integrated within the GUHRI system.

8 Conclusions

The GUHRI system, a novel body gesture understanding and human-robot interaction system, is proposed in this paper. A set of 12 human upper body gestures with and without human-object interactions can be understood by the robot. Meanwhile, the robot can express herself by using a combination of body movements, facial expressions and verbal language simultaneously, aiming to give the users vivid experience.

A new combination of sensors is proposed. That is, the CyberGlove II and Kinect are combined to capture the head, arm and hand posture simultaneously. An effective and real-time gesture recognition method is also proposed. For robot control, a novel online motion planning method is proposed. This motion planning method is formulated as a quadratic program style. It is then solved by a simplified recurrent neural network. The obtained optimization solutions are finally used to generate arms movements. This method has been integrated within a robot controller module. In experiment, a human upper body gesture dataset is built. The experimental results demonstrate the effectiveness and efficiency of our gesture recognition method and motion planning approach.

So far, the gestures involved in GUHRI system are static ones, e.g., *"have question"*, *"praise"*, *"call"* and *"drink"*, etc. As future work, we plan to enable the robot to understand the dynamic gestures, such as *"wave hand"*, *"type keyboard"* and *"clap"*, etc. Speech recognition can be further added to make the interaction more natural.

Acknowledgements

This work partially carried out at BeingThere Center is supported by IMI Seed Grant M4081078.B40, and Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.
This work is an extended and improved version of Xiao, Y., Yuan, J., & Thalmann, D. (2013), Human-virtual human interaction by upper body gesture understanding. To be found in Proceedings of the 19th ACM symposium on virtual reality software and technology (VRST 2013) (pp. 133–142).

Appendices

The demonstration video of this paper can be viewed at https://www.dropbox.com/s/eht69b8kyg53vso/hri_demo_mit.mp4.

References

Adamo-Villani, N., Heisler, J., & Arns, L. (2007). Two gesture recognition systems for immersive math education of the deaf. In *Proceedings of the first international conference on immersive telecommunications (ICIT 2007)* (p. 9). 6

Beck, A., Cañamero, L., Hiolle, A., Damiano, L., Cosi, P., Tesser, F., & Sommavilla, G. (2013). Interpretation of emotional body language displayed by a humanoid robot: A case study with children. *International Journal of Social Robotics*, 1–10. 3

Beck, A., Hiolle, A., & Cañamero, L. (2013). Using perlin noise to generate emotional expressions in a robot. In *Proceedings of annual meeting of the cognitive science society (COG SCI 2013)* (p. 1845-1850). 20

Beck, A., Stevens, B., Bard, K. A., & Cañamero, L. (2012, March). Emotional body language displayed by artificial agents. *ACM Transactions on Interactive Intelligent Systems*, *2*(1), 2:1–2:29. 7

Belongies, S., Malik, J., & Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(4), 509–522. 17

Berman, S., & Stern, H. (2012). Sensors for gesture recognition systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *42*(3), 277–290. 6

Brethes, L., Menezes, P., Lerasle, F., & Hayet, J. (2004). Face tracking and hand gesture recognition for human-robot interaction. In *Proceedings of IEEE conference on robotics and automation (ICRA 2004).* (Vol. 2, pp. 1901–1906). 5

Cai, B., & Zhang, Y. (2012). Different-level redundancy-resolution and its equivalent relationship analysis for robot manipulators using gradient-descent and zhang 's neural-dynamic methods. *IEEE Transactions on Industrial Electronics*, *59*(8), 3146-3155. 8

Cañamero, L., & Fredslund, J. (2001). I show you how i like you - can you read it in my face? [robotics]. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, *31*(5), 454-459. 25

Cassell, J., et al. (2000). Nudge nudge wink wink: Elements of face-to-face conversation for embodied conversational agents. *Embodied conversational agents*, 1–27. 3

Chan, T. F., & Dubey, R. (1995). A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators. *IEEE Transactions on Robotics and Automation*, *11*(2), 286-292. 8

Cheng, F.-T., Chen, T.-H., & Sun, Y.-Y. (1994). Resolving manipulator redundancy under inequality constraints. *IEEE Transactions on Robotics and Automation*, *10*(1), 65-71. 8

Chopra, S., Hadsell, R., & LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of IEEE conference on computer vision and pattern recognition (CVPR 2005)* (Vol. 1, pp. 539–546). 16

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transtractions in Information Theory*, *13*(1), 21–27. 17

Dautenhahn, K. (2007). Socially intelligent robots: dimensions of human–robot interaction. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *362*(1480), 679–704. 2,

3

Faber, F., Bennewitz, M., Eppner, C., Gorog, A., Gonsior, C., Joho, D., ... Behnke, S. (2009). The humanoid museum tour guide robotinho. In *Proceedings of IEEE symposium on robot and human interactive communication (RO-MAN 2009)* (pp. 891–896). 3, 6

Fisher, R. A. (1936). The use of multiple measures in taxonomic problems. *Annals of Eugenics*, *7*, 179-188. 28

Fong, T., Nourbakhsh, I., & Dautenhahn, K. (2003). A survey of socially interactive robots. *Robotics and autonomous systems*, *42*(3), 143–166. 2, 3

Goodrich, M. A., & Schultz, A. C. (2007). Human-robot interaction: a survey. *Foundations and Trends in Human-Computer Interaction*, *1*(3), 203–275. 2

Guo, D., & Zhang, Y. (2012). A new inequality-based obstacle-avoidance mvn scheme and its application to redundant robot manipulators. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *42*(6), 1326-1340. 8

Immersion. (2010). *Cyberglove II specfications.* Retrieved from http://www.cyberglovesystems.com/products/cyberglove-ii/specifications 4, 14, 38

Jolliffe, I. T. (1986). *Principal component analysis*. Springer-Verlag. 28

Kanoun, O., Lamiraux, F., & Wieber, P. B. (2011). Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task. *IEEE Transactions on Robotics*, *27*(4), 785-792. 8

Krämer, N. C., Tietz, B., & Bente, G. (2003). Effects of embodied interface agents and their gestural activity. In *Intelligent virtual agents* (pp. 292–300). 25

Lu, G., Shark, L.-K., Hall, G., & Zeshan, U. (2012). Immersive manipulation of virtual objects through glove-based hand gesture interaction. *Virtual Reality*, *16*(3), 243–252. 4, 6

Ma, S., & Watanabe, M. (2002). Time-optimal control of kinematically redundant manipulators with limit heat characteristics of actuators. *Advanced Robotics*, *16*(8), 735-749. 8

Martins, A., Dias, A., & Alsina, P. (2006). Comments on manipulability measure in redundant planar manipulators. In *Proceedings of IEEE latin american robotics symposium (LARS*

2006) (p. 169-173). 8

Mehrabian, A. (1971). Silent messages.

3

Miyashita, T., & Ishiguro, H. (2004). Human-like natural behavior generation based on involuntary motions for humanoid robots. *Robotics and Autonomous Systems*, *48*(4), 203 - 212. 18

Müller, M., & Röder, T. (2006). Motion templates for automatic classification and retrieval of motion capture data. In *Proceedings of the 2006 ACM siggraph/eurographics symposium on computer animation (SCA 2006)* (pp. 137–146). 17

Nickel, K., & Stiefelhagen, R. (2007). Visual recognition of pointing gestures for human–robot interaction. *Image and Vision Computing*, *25*(12), 1875–1884. 3

Osborne, J. W., & Costello, A. B. (2004). Sample size and subject to item ratio in principal components analysis. *Practical Assessment, Research & Evaluation*, *9*(11), 8. 29

Perzanowski, D., Schultz, A. C., Adams, W., Marsh, E., & Bugajska, M. (2001). Building a multimodal human-robot interface. *IEEE Intelligent Systems*, *16*(1), 16–21. 3

Pierris, G., & Lagoudakis, M. (2009). An interactive tool for designing complex robot motion patterns. In *Proceedings of IEEE international conference on robotics and automation (ICRA 2009)* (p. 4013-4018). 18

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., ... Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In *Proceedings of IEEE conference on computer vision and pattern recognition (CVPR 2011)* (pp. 1297–1304). 4, 6, 14

Smith, L. B., & Breazeal, C. (2007). The dynamic lift of developmental process. *Developmental Science*, *10*(1), 61–68. 3

Spiliotopoulos, D., Androutsopoulos, I., & Spyropoulos, C. D. (2001). Human-robot interaction based on spoken natural language dialogue. In *Proceedings of the european workshop on service and humanoid robots* (pp. 25–27). 3

Stiefelhagen, R., Ekenel, H. K., Fugen, C., Gieselmann, P., Holzapfel, H., Kraft, F., ... Waibel, A. (2007). Enabling multimodal human–robot interaction for the karlsruhe humanoid robot.

IEEE Transactions on Robotics, *23*(5), 840–851. 3, 5

Stiefelhagen, R., Fugen, C., Gieselmann, R., Holzapfel, H., Nickel, K., & Waibel, A. (2004). Natural human-robot interaction using speech, head pose and gestures. In *Proceedings of IEEE conference on intelligent robots and systems (IROS 2004)* (Vol. 3, pp. 2422–2427). 3, 5

Taghirad, H. D., & Nahon, M. (2008). Kinematic analysis of a macromicro redundantly actuated parallel manipulator. *Advanced Robotics*, *22*(6-7), 657-687. 8

Takahashi, Y., Kimura, T., Maeda, Y., & Nakamura, T. (2012). Body mapping from human demonstrator to inverted-pendulum mobile robot for learning from observation. In *Proceedings of IEEE conference on fuzzy systems (FUZZ-IEEE 2012)* (p. 1-6). 18

Teleb, H., & Chang, G. (2012). Data glove integration with 3d virtual environments. In *Proceeedings of international conference on systems and informatics (ICSAI 2012)* (pp. 107–112). 4

Waldherr, S., Romero, R., & Thrun, S. (2000). A gesture based interface for human-robot interaction. *Autonomous Robots*, *9*(2), 151–173. 5

Wang, J., & Li, Y. (2009). Inverse kinematics analysis for the arm of a mobile humanoid robot based on the closed-loop algorithm. In *Proceedings of international conference on information and automation (ICIA 2009)* (p. 516-521). 7

Wang, J., Liu, Z., Wu, Y., & Yuan, J. (2012). Mining actionlet ensemble for action recognition with depth cameras. In *Proceedings of IEEE conference on computer vision and pattern recognition (CVPR 2012)* (pp. 1290–1297). 6, 15

Weinberger, K. Q., & Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, *10*, 207–244. 4, 11, 16, 18, 27

Xiao, Y., Cao, Z., & Yuan, J. (2014). Entropic image thresholding based on GLGM histogram. *Pattern Recognition Letters*, *40*, 47–55. 6

Xiao, Y., Wu, J., & Yuan, J. (2014). mCENTRIST: A multi-channel feature generation mechanism for scene categorization. *IEEE Transactions on Image Processing*, *23*(2), 823–836. 6

Xiao, Y., Yuan, J., & Thalmann, D. (2013). Human-virtual human interaction by upper body gesture understanding. In *Proceedings of the 19th ACM symposium on virtual reality software and technology (VRST 2013)* (pp. 133–142). 6

Zhang, Y., Tan, Z., Yang, Z., Lv, X., & Chen, K. (2008). A simplified LVI-based primal-dual neural network for repetitive motion planning of pa10 robot manipulator starting from different initial states. In *Porceedings of IEEE joint conference on neural networks (IJCNN 2008)* (p. 19-24). 24

Zhang, Y., Wu, H., Zhang, Z., Xiao, L., & Guo, D. (2013). Acceleration-level repetitive motion planning of redundant planar robots solved by a simplified lvi-based primal-dual neural network. *Robotics and Computer-Integrated Manufacturing*, *29*(2), 328 - 343. 24

Zhang, Z., & Zhang, Y. (2012). Acceleration-level cyclic-motion generation of constrained redundant robots tracking different paths. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, *42*(4), 1257-1269. 8

Zhang, Z., & Zhang, Y. (2013a). Design and experimentation of acceleration-level drift-free scheme aided by two recurrent neural networks. *Control Theory & Applications, IET*, *7*(1), 25-42. 24

Zhang, Z., & Zhang, Y. (2013b). Equivalence of different-level schemes for repetitive motion planning of redundant robots. *Acta Automatica Sinica*, *39*(1), 88-91. 24

Zhang, Z., & Zhang, Y. (2013c). Variable joint-velocity limits of redundant robot manipulators handled by quadratic programming. *IEEE/ASME Transactions on Mechatronics*, *18*(2), 674-686. 8, 22, 23

List of Figures