Smartphone-based Human Machine Interface with Application to Remote Control of Robot Arm

Carlos Román Parga Villalpando
Departamento de Computacin
CINVESTAV-IPN
Mxico City, 07360, Mxico

Xiaoou Li
Departamento de Computacin
CINVESTAV-IPN
Mxico City, 07360, Mxico

Wen Yu
Departamento de Control Automtico
CINVESTAV-IPN
Mxico City, 07360, Mxico

***Abstract*—In order to design a small size, cheap, space sensing, and wireless communication human machine interface (HMI), in this paper we develop a smartphone-based system. It uses the accelerometer and the gyroscope of the smartphone to generate six commands. By calculating the inverse kinematics, the reference angles are send to robot arm via Wi-Fi network. The novel HMI has many advantages over the others. The experiment results show that our HMI is convenient and effective for remote control of robot arm.**

I. Introduction

Human Machine Interface (HMI) is regarded as a system that handles the human machine interaction. It, more commonly Human Computer Interaction (HCI) [3], is the study of interaction between machine (or computer) and people [1]. HMI can be regarded as part works of Human Robot Interaction (HRI) [2]. The human machine interface system efficiently integrates human into an automation system. It has two components: a human user uses it to send commands to the machine, and the machine shows its running status to the human user via the human machine interface system. A good human machine interface can reduce the risk of human's injury, fatigue, and discomfort, and improve productivity and the quality of the interaction. There are many human machine interface technologies, such as vision (light) based [4], acoustic based [5], brain computer interfacing [6], electromyography (EMG) based [7], tactile technology [8], and motion technology [9]. Most of human robot arm interfaces use EMG, tactile and motion sensors.

Many human-like robot arm, such as robot exoskeletons, have more than six degrees freedom [10]. They are three translations in the perpendicular axes and three rotations about these perpendicular axes (pitch, yaw, and roll). EMG (Electro-Myo-Graphic), ENG (Electro-Neuro-Graphic) or even EEG (Electro-Encephalo-Graphic) measurements could possibly provide several motion commands for robots [11]. However, they suffer from important drawbacks: 1) it is difficult to obtain the same EMG signals for the same motion even with the same person; 2) it is not easy to predict real time motion since many muscles are involved in a joint motion; 3) the role of each muscle for a certain motion varies with joint angles. The normal 2D joystick or 3D joystick [12] can send two or three position commands. The joystick cannot send six commands simultaneously. In order to generate rotation commands, a six-axis force/torque sensor is applied in [13]. However the sensor and its interface are very expensive, and the position and rotation signals cannot be obtained directly from the force/torque sensor. All of the above sensors need computers to be the human machine interface systems. Most of them cannot be used as remote controllers.

The human machine interface for robot arm remote control needs the following properties: the size (computer with sensors) is small; it is not expensive; it can send at least six commands from different sensors; it has wireless communication. There is a device satisfies all these requirements, it is the *smartphone*. The use of smartphone in robotics interface is a relatively new concept [14], since the introduction of the smartphone is less than eight years. The main attraction of the smartphone is its platform. It has a fast CPU, memory, a monitor, common operating system and rich sensors. The programming is very similarly as a conventional PC. Unlike the laptop, the smartphone can always be operated with the battery and is lightweight. Many smartphones have compass, accelerometer, gyroscope, thermometer, light meter, camera, and GPS receiver. They also have standard wireless communication devices, such as Wi-Fi, Bluetooth and 3G. The smartphone as the human machine interface has two types tasks:

- Monitoring. The smartphone is used as a computer to display and store data of the robot [16][15][17].
- Control. The touch screen and keyboard smart phones are used to send the robot commands [18][19][20].

The first generation of smartphone interface can be regarded as using a PDA as user interface [21]. It use Wi-Fi infrastructure to send commands from PDA touch screen, and monitoring the mobile robot from PDA screen. NASA applied the smartphone operating system, Android, in a prototype of SPHERES for the navigation and docking system of International Space Station [22]. By using the accelerometer of the smartphone, human hand gestures become control commands for the mobile robot [23]. In the high risk area, the mobility and small size properties of the smartphone have great advantage over the other interfaces [24]. Another advantage of the smartphone interface is the operations are very simple via the touch screen and sensors of the smartphone [25].

There are still problems of the smartphone interface.

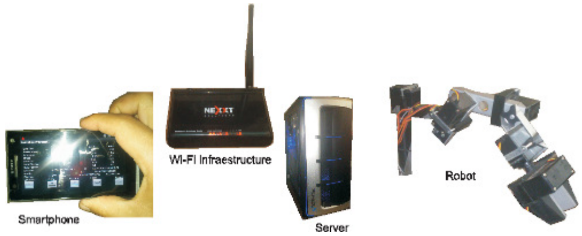

Fig. 1. The overview of HMI.

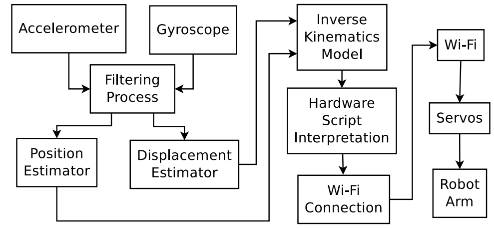

Fig. 2. The workflow of the HMI.

1) Although the small size of the smartphone is comfortable for human hand movement, small buttons in the touch screen are not convenient. A full sensors-based smartphone interface is needed.
2) The accuracy of the smartphone sensors is not in industrial level. It is impossible to realize precision control with the smartphone.
3) The position sensor of the smartphone, GPS, cannot be used for robot arm control. Although double integration of the accelerometer generates the position signals, the error is very big with the offset problem.
4) The smartphone in the human machine interface system has to complete both jobs of the computer and interface. It requires more effective software.

Another difficult of controlling robot arm via the smartphone is the robot arm need joint space commands, while the smartphone works in task space. The inverse kinematics of the robot arm has to be calculated in the smartphone.

In this paper, we develop a human machine interface by using the accelerometer and gyroscope sensors of the smartphone. The accelerometer gives three positions and the gyroscope provides three rotations. The offset in the position estimation is canceled by the gyroscope signals and calibration. The remote control only requires relative positions, the sensor errors can be decreased by human vision feedback.

In order to maintain small control loop time (sensor filter, robot kinematics, communication and interpretation of the commands), a simplified inverse kinematics algorithm is proposed in this paper. Finally, we apply our smartphone interface to a five degree freedom robot. Experimental results validates our design.

II. System Hardware Architecture

A. System overview

An overview of the system is shown in Fig.1. It includes three subsystems:

- Smartphone platform. This is main part of our HMI. It uses accelerometers and gyroscopes in the smartphone to catch the behavior of human sensors. By using some algorithms, the human action is transferred into three positions (x, y, and z) and three rotations (yaw, pitch, and roll). These task space commands are transformed into joint space signals by a simplified inverse kinematic algorithm. Finally the reference joint angles are sent. The smartphone subsystem also has graphic user interface (GUI).
- Wireless local area network. This subsystem includes a Wi-Fi router, the Wi-Fi interface in the smartphone, and the Wi-Fi interface, such as USB Wi-Fi adaptor, for the robot arm controller. The communication protocol is Transmission Control Protocol (TCP). We use IEEE802.11n standard. The wireless range is 60 m indoors. The frequency band is 2.4 GHz.
- Robot arm. This system has robot arm and servo controllers for the joints of the robot. We use the standard proportional-integral-derivative (PID) for the joint servo control. Each robot servo controller receives the reference joint angle from the smartphone, and force the joint motor to follow the reference angle by PID control.

The workflow of the above three subsystems is shown in Fig.2.

B. Smartphone platform

The hardware of smartphone offers many advantages in peripheral sensing, central processing and wireless communication. Most of smartphones have Wi-Fi communication module, accelerometer and gyroscope. The average CPU speed of the smartphone is currently 1GHz. The normal memory of the smartphone is higher than 256Mb. These features allow the smartphone to perform like a personal computer with several special sensors. It is easy to carry out some software application, such as filtering and inverse kinematic, in the smartphone.

The smartphone-based HMI uses two sensors: accelerometer (m/s^2) and gyroscope (rad/s). The accelerometer inside the smartphone gives three acceleration of the smartphone, see Fig.3. The three axes are defined as follows: X and Y are parallel to the two sides of the the displace screen. Z axis is perpendicular to the screen. The origin of the axes is the center of the screen.

The gyroscope of the smartphone gives three rotation directions allow the axes of the accelerometer, see Fig.4. The clockwise direction is defined as negative, while counter clockwise is positive.

There two sensors are used to estimate relative displacements (or positions) and directions of the smartphone. The workflow of the sensors of the smartphone is shown in Fig.5.

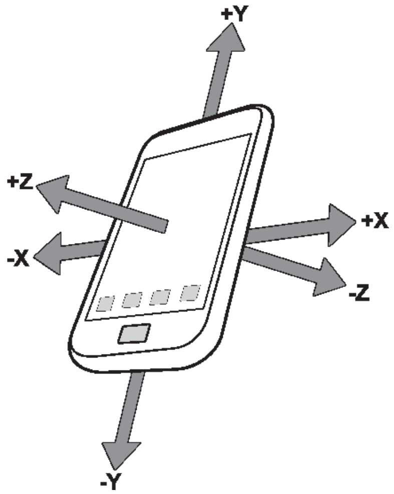

Fig. 3. Accelerometer of the smartphone

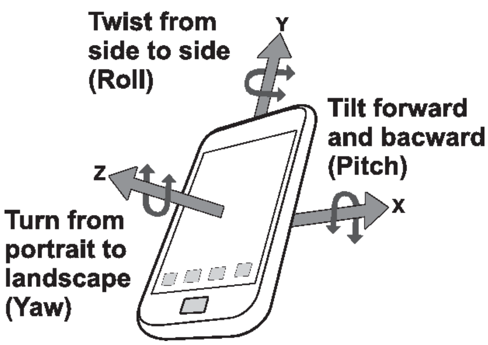

Fig. 4. Gyroscope of the smartphone

C. Robot arm

In order to move a robot arm, a servo controller and a computer are needed. The computer is used to run control software, such as PID control, and communication software. The servo controller is connected to the robot arm directly and receives control command from the computer.

In our example, the servo controller of Robix™ robot arm [27] is connected to the computer via USB. The computer sends reference joint angles to the servo controller. the PID algorithm is realized in the servo controller. In this case, the computer is only a communication interface. If a servo controller has Wi-Fi module, the computer can be removed from the system.

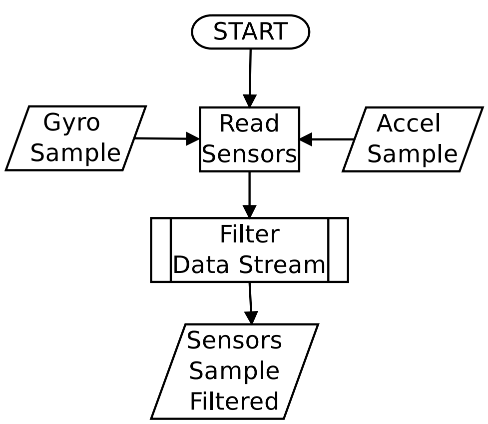

Fig. 5. Workflow of the sensors.

The advantage of using a computer is the control software can be on Windows platform, and the configuration is easy using the drivers and libraries for Java™ provided from Robix™ Software.

D. Wireless local area network

The wireless infrastructure enables communication between the smartphone and the robot. The advantage is that both components communicating with the TCP protocol regardless of the medium through physical protocol structure. In this way is transparent equipment you use is for communication and the means by allowing travel information using communication LAN without changing the software. The advantage you have is that commercial Wi-Fi equipment are cheap and high transfer speeds so that it is not necessary be very careful in choosing the type of equipment used always they comply with the IEEE802.11 standard for the smartphone connect to the network and communicate.

III. Software Architecture

In our application, the operating system of the smartphone is chosen Android, because it is open source and many smartphones in the market use it. Another advantage is that all internal peripheral devices can be accessed directly by using appropriate libraries in Android. For example, the data from the accelerometer and gyroscope in the smartphone can be read in Android operating system via a sensor manager service which is configured to obtain data at regular intervals.

The software includes five modules: signal filtering, position and direction computation, inverse kinematics, communication, and graphic user interface. These modules are carried out in each control loop. Fig.6 shows the main function in the control loop. The initial position of the robot arm is set to zero. This initial position will be used in the calculation of the inverse kinematics.

A. Signal filtering

The signals from the accelerometer and gyroscope sensors of the smartphone are raw data which include noises and offsets. The noises come from sensor drift and hand arm vibration. To avoid mechanical vibration of the robot arm, these noises must be eliminated. Since most of noises appear in high frequency, we use a low pass finite impulse response (FIR) filter

$$\frac{y(k)}{x(k)} = \sum_{i=0}^{n} b_i z^{n-i} \tag{1}$$

where k is the sampling time, $y(z)$ is the output signal, $x(z)$ is the input signal, b_i are the filter coefficients, n is the filter order, the Z. transformation has the property of $x(k-i) = x(k)z^{-i}$.

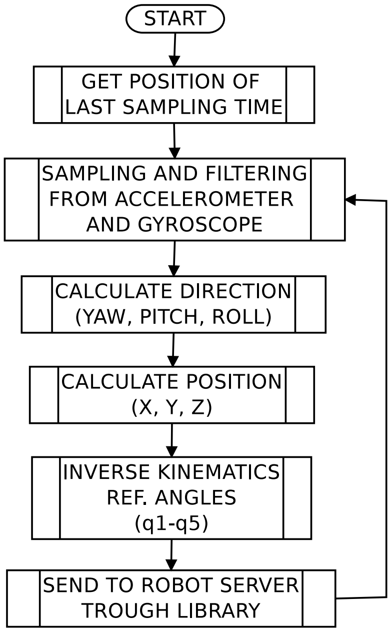

Fig. 6. Control loop

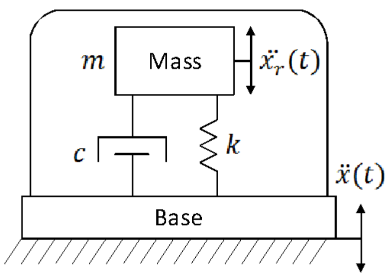

Fig. 7. Mechanical model of an accelerometer

The FIR filter does not require an infinite impulse response. In our application, we found when the cutoff frequency is $4Hz$ and the filter order is 2, the low pass filter (1) has good property with respect to hand arm vibration. The filter coefficients b_i are obtained by Hamming window method [28].

The ideal order of the filter should be infinite in general. A larger order results in a better approximation. On the other hand, a higher order filter involves more computation cost in the smartphone.

Because the noise sources for the accelerometer and the gyroscope are the same (user's hand), we use the same filter for the three axes of the accelerometer and the gyroscope., i.e., we use six FIR filters.

B. Position calculation

The accelerometer can be regarded as a single-degree-of-freedom mechanical system [29]. The inertial force F acting on the proof-mass m is given by

$$F = m(\ddot{x}(t) + \ddot{x}_r(t)) \tag{2}$$

where $\ddot{x}(t)$ is the acceleration acting on the accelerometer, $\ddot{x}_r(t)$ is the relative acceleration of the proof-mass with respect to the base. The dynamics of the accelerometer subjected to acceleration $\ddot{x}(t)$ using Newton's second law is given by

$$m\ddot{x}_r(t) + c\dot{x}_r(t) + kx_r(t) = -m\ddot{x}(t) \tag{3}$$

where k is the stiffness of the spring, c is the damping coefficient, see Fig.7. The deflection due to the acceleration is sensed and converted into an equivalent electrical signal as

$$\ddot{x}_r(t) + 2\zeta\omega\dot{x}_r(t) + \omega^2 x_r(t) = k_a\ddot{x}(t) \tag{4}$$

where k_a is the accelerometer gain, ζ is the system damping constant, and ω is the resonant frequency, here $\zeta = \frac{c}{2\sqrt{km}}$, and $\omega = \sqrt{\frac{k}{m}}$. The output of the accelerometer is

$$a(t) = k_a\ddot{x}(t) + w(t) + d \tag{5}$$

where $w(t)$ is the noise and disturbance, d denotes the zero gravity offset ($0g$-offset). The noise $w(t)$ is minimized by the low pass filter proposed in (1).

The $0g$-offset is measured under the absence of motion or gravity. This offset may vary from one sensor to another. The main causes of this variation are the sensing material, temperature, supply voltage deviation, mechanical stress and trim errors. The knowledge of this offset error will assist in removing the bias from the acceleration signal effectively. We use calibration method to find d.

Mathematically the velocity $\dot{x}(t)$ and position $x(t)$ are calculated by integrating the acceleration $\ddot{x}(t)$

$$\begin{aligned} \dot{x}(t) &= \int_0^t \ddot{x}(\tau)d\tau + \dot{x}(0) \\ x(t) &= \int_0^t \int_0^\tau \ddot{x}(\tau)d\tau dt + \dot{x}(0)t + x(0) \end{aligned} \tag{6}$$

where $\dot{x}(0)$ and $x(0)$ is the initial velocity and position, respectively.

If the accelerometer is not mounted on the ground, the gravitational acceleration should be taken into account. The output of the accelerometer becomes

$$a(t) = k_a\ddot{x}(t) + w(t) + d + g \tag{7}$$

When the smartphone has yaw α, pitch β, and roll γ angles, the acceleration outputs in each axis are

$$\begin{aligned} a_x(t) &= k_a\ddot{x}_x(t) + w_x(t) + d_x + g\cos\gamma \\ a_y(t) &= k_a\ddot{x}_y(t) + w_y(t) + d_y + g\cos\beta \\ a_z(t) &= k_a\ddot{x}_z(t) + w_z(t) + d_z + g\cos\alpha \end{aligned} \tag{8}$$

where α, β, γ are measured by the gyroscope of the smartphone, $\ddot{x}_x(t)$, $\ddot{x}_y(t)$, $\ddot{x}_z(t)$ are applied to the double integrator (6).

In the robot arm control, we only need to send the relative position to the robot. The absolute position of the smartphone

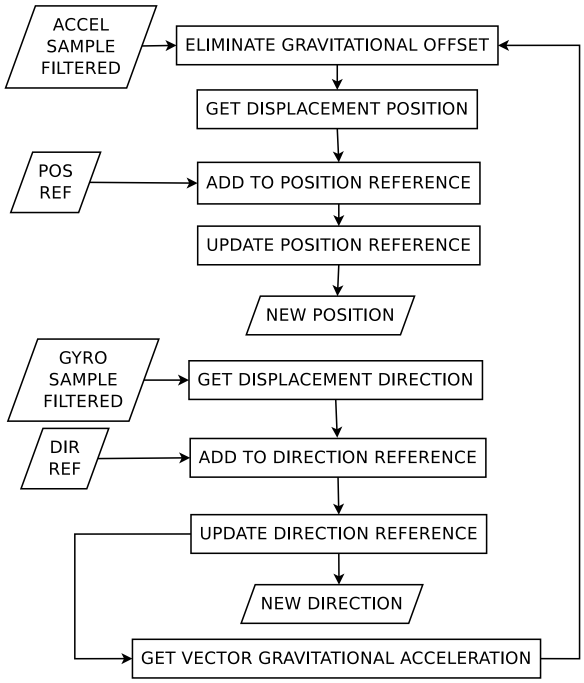

Fig. 8. Position calculation

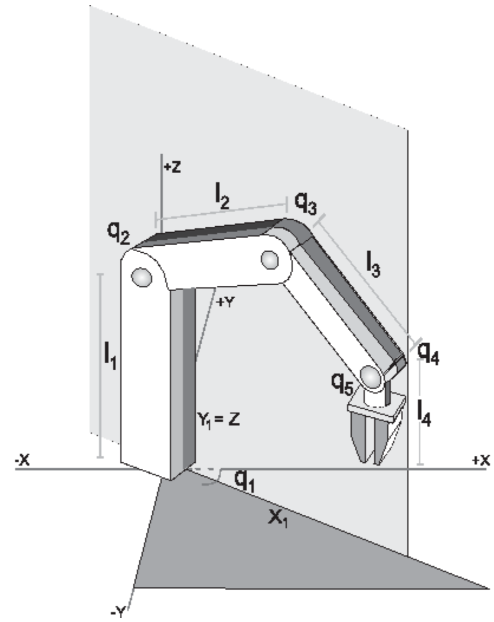

Fig. 9. 5-DOF robot arm.

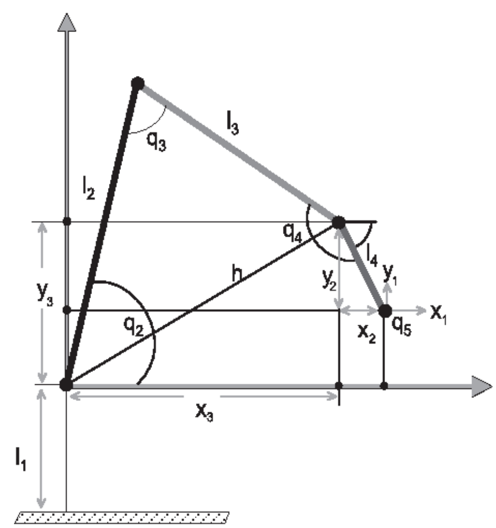

Fig. 10. Trigonometric cinematic Y Z axis

does not have sense. If we move the smartphone from position A to position B, and let the initial velocity $\dot{x}(0) = 0$

$$x_B(t) - x_A(t) = \int_0^{t_B}\int_0^{\tau} \ddot{x}(\tau)d\tau dt - \int_0^{t_A}\int_0^{\tau} \ddot{x}(\tau)d\tau dt$$
$$= \int_{t_A}^{t_B}\int_0^{\tau} \ddot{x}(\tau)d\tau dt \tag{9}$$

We use (9) to calculate the relative positions the robot arm should be moved

$$\Delta x = \int_{t_A}^{t_B}\int_0^{\tau} \ddot{x}_x(\tau)d\tau dt, \Delta y = \int_{t_A}^{t_B}\int_0^{\tau} \ddot{x}_y(\tau)d\tau dt$$
$$\Delta z = \int_{t_A}^{t_B}\int_0^{\tau} \ddot{x}_z(\tau)d\tau dt \tag{10}$$

In discrete-time, the numerical integration is performed to get an approximation by applying the numerical interpolation,

$$\int_{t_0}^{t_n} \ddot{x}(t)dt \approx \sum_{i=1}^{n} \left[\frac{\ddot{x}(i-1) + \ddot{x}(i)}{2}\right] \Delta t \tag{11}$$

Fig.8 shows the position calculation process.

C. Inverse kinematics

The relative positions in (10) cannot be used to control the joint motors of the robot arm directly. The position information should be transformed into reference angles as

$$q = K^{-1}\left(\Delta x, \Delta y, \Delta z\right), \quad q \in R^n \tag{12}$$

where the robot arm has n degree-of-freedom (DOF). K^{-1} is the inverse kinematics, it is a function from the end-effector position $((x, y, z))$ to the joint angle $(q_1 \cdots q_n)$. Usually when $n > 3$, K^{-1} is not unique.

Kinematic decoupling and geometric approach are the most popular methods [30]. Although a complex robot arm needs 6-DOF, the 5-DOF robot arm has been applied in many areas which is more simple than the 6-DOF robot and only loses 1-DOF in the end-effector [31]. In this paper we use a 5-DOF robot arm to show how our smartphone-based HMI works, see Fig.10.

In order to reduce computation burden in the smartphone, we propose a simplified geometric approach for the inverse kinematics. The end-effector position id defined as (x, y, z).

The first is to calculate the angle of rotation of the arm as show in figure9.

From Fig.10 we known

$$q_1 = \arctan\left(\frac{x}{y}\right)$$

where q_1 angle of Joint-1. Then you make a change of variables to calculate the rest of the joints in two dimensions where using only two axes (x_1, y_1) and two position (pitch $->p_x$, roll $->p_z$)

$$\begin{aligned} x_1 &= \sqrt{x^2 + y^2} \\ y_1 &= z \end{aligned}$$

In this manner the rest of the joint angles in two axes is calculated as follows with reference to the Fig.10:

$$\begin{aligned} x_2 &= l_4 \cos(p_x) \\ x_3 &= x_1 - x_2 \\ y_2 &= l_4 \sin(p_x) \\ y_3 &= y_1 - y_2 - l_1 \\ h &= \sqrt{x_3^2 + y_3^2} \end{aligned}$$

Now you have all the values needed to calculate the angles of all joints as follows

$$\begin{aligned} q_1 &= \arctan\left(\tfrac{x}{y}\right) \\ q_2 &= \arctan(\tfrac{y_3}{x_3}) + \cos^{-1}(\tfrac{l_2^2 - l_3^2 + h^2}{2hl_2}) \\ q_3 &= \pi - \cos^{-1}(\tfrac{l_2^2 + l_3^2 - h^2}{2l_2 l_3}) \\ q_4 &= p_x - q_2 - q_3 \\ q_5 &= p_z \end{aligned}$$

The above simplified procedure finishes the job in (12).

D. Communication

The inverse kinematics module generates the reference joint angle $(q_1 \cdots q_n)$. These reference values are interpreted into script language and send to the robot arm servo controller. The communication is through the call of servo library. The robot servo controller interprets the script, and creates own movement sequences by PID algorithm.

We use TCP/IP communication protocol between the remote client and the host smartphone. The client receives script sequences from the host according to it own IP address. The client uses a Java library to explain received ASCII strings

The robot arm connects to Internet with an unique IP address. The smartphone also connect to Internet via a Wi-Fi router. So our smartphone-based HMI is not only a wireless system, but also is a remote controller for tele-operation via Internet.

E. Graphic user interface

The graphic user interface of the smartphone has two tasks: sending simple commands and monitoring the behaviors of the robot arm.

There is a START button to tell the smartphone to start the application and establish communication with the server. We design another button to tell the smartphone there is a new movement beginning the process of estimating position and displacement. Such that only one motion script can be sent each time.

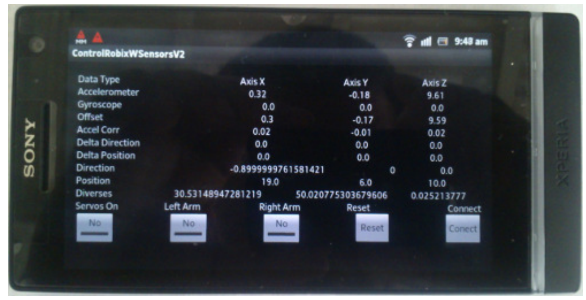

Fig. 11. User interface monitoring

Fig. 12. Phisical System Overview

The monitoring system displays the running status of the HMI, see Fig.11. It display real-time data of all sensors, the control commands for the robot arm and the behavior of the robot.

IV. Experimental Results

In order to demonstrate the smartphone-based HMI proposed in this paper, we developed a prototype system shown in Fig.1. The smartphone is SONY Xperia S with Android 2.3, 1.5 GHz Qualcomm Dual Core processor, and $1G$ RAM. The sensors of this phone include an accelerometer, an electromagnetic field sensor (proximity sensor), an ambient light sensor, a magnetometer, a GPS sensor, and a gyroscope. The router is Next Nebula 150 with 4 10/100 Mbps LAN ports and Wireless IEEE802.11n ports. The 5-DOF is constructed with Robix parts [27], see Fig.12.

We designed two types of tests: linear and circular motions, to validate our HMI.

A. Linear motion

In this test, the smartphone is mounted on a small rail cart, and is moved from point A to point B, see Fig.13. The distance of $A - B$ can be measured.

This test includes two parts: the first part evaluates only the estimation error of the accelerometer and the gyroscope. The results are shown in Table I

Fig. 13. Linear Motion Test

Measure	Estimated	Error
65	79.5	14.5
65	74.7	9.7
65	75.9	10.9
65	73.5	8.5
65	66	1
65	71.4	6.4
65	77.4	12.4
65	66.6	1.6
65	73.5	8.5
65	69.3	4.3

TABLE I
ESTIMATION ERROR OF OF THE ACCELEROMETER AND THE GYROSCOPE

The second part check the communication and inverse kinematics. The results are shown in Table II..

Measure	Estimated	Displacement	Error
9	8.3	9	0
9	9.7	10.7	1.7
9	7.5	7.5	-1.5
9	8.7	8.7	-0.3

TABLE II
ESTIMATION ERRORS OF THE COMMUNICATION AND INVERSE KINEMATICS

B. Circle motion

In this test, we use the smartphone draw a circle, and send the corresponding commands to the robot arm. The robot should also dram a similar circle. Similar as the linear motion, the test also includes two parts. The results of the communication and inverse kinematics test are shown in Fig.14, here the circle is generated mathematically from the smartphone.

The results of the second part are shown in Fig.15, where the circles are drawn by the robot arm.

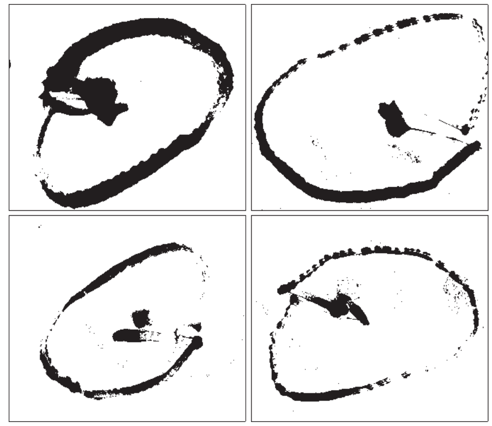

Fig. 14. Circle motion of the robot arm

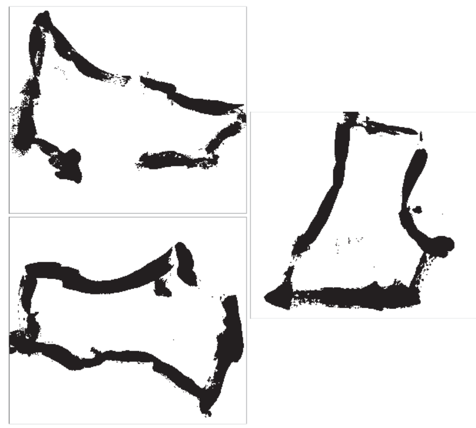

Fig. 15. Circle Motion Inertial Results

C. Discussions

Response Time

The time from the smartphone moving to the end-effector of the robot arm reaching the desired point depends on computing time of the smartphone, communication network delay, and robot servo time. The network traffic is the most uncertainty. We consider the worse case of it. We use 1 second for each control loop in Fig.6.

Sensitivity Resolution

The accelerometer of the smartphone cannot capture any human movement. There exist the maximum and the minimum speeds. These speeds are also effected by the offset and the noise filter.

Estimation error

The estimation error (Table I) is approximately 12% and the total error of 9.7%. These errors come from the sensor error and noises. The HMI system doe not have position feedback. The PID control in robot servo can only force each motor follow the desired angle. The simplified inverse kinematics also causes error.

V. CONCLUSION

In this paper a novel human machine interface is proposed. It uses a smartphone to finish the following jobs: position measurement, rotation capture, inverse kinematics calculation, Wi-Fi communication, and GUI. Our smartphone-based HMI overcome the common problems of the other types of HMI, such as big size, expensive, one device with six commands, and wireless communication.

Compared with the other smartphone interface, the advantages of our system are: 1) we use both accelerometer and gyroscope to capture positions and rotations. It is a space sensing method; 2) Signal filtering and gravity compensation are used yo improve the sensor accuracy; 3) Many algorithms are simplify to reduce computation cost. The experiment results show that our HMI is convenient and effective for remote control of robot arm.

References

[1] Denis Lalanne, Juerg Kohlas (Eds.), *Human Machine Interaction-Research Results of the MMI Program*, Lecture Notes in Computer Science, Vol.5440, Springer, 2009

[2] Michael A. Goodrich and Alan C. Schultz, Human Robot Interaction: A Survey, *Foundations and Trends in Human Computer Interaction*, Vol. 1, No. 3, 203-275, 2007

[3] Alan Dix, Janet Finlay, Gregory Abowd, and Russell Beale, *Human–Computer Interaction,* 3rd Edition. Prentice Hall, 2003

[4] Branislav Kisacanin, Vladimir Pavlovic, Thomas S. Huang, *Real-Time Vision for Human-Computer Interaction*, Springer, 2005

[5] Helena Roeber, John Bacus, Carlo Tomasi, Typing in thin air: the canesta projection keyboard - a new method of interaction with electronic devices, *CHI '03 Human Factors in Computing Systems*, 712-713 2003

[6] D.J.McFarland, J.R.Wolpaw, Brain-computer interface operation of robotic and prosthetic devices, *Computer*, Vol.40, No.10, 52-56, 2008,

[7] Andrzej Wołczowski, Marek Kurzynski, Human–machine interface in bioprosthesis control using EMG signal classification, *Expert Systems,* Volume 27, Issue 1, pages 53–70, 2010

[8] Steven A Wall, William Harwin, A high bandwidth interface for haptic human computer interaction, *Mechatronics*, Volume 11, Issue 4, Pages 371–387, 2001

[9] Cheng-Chang Lien, Chung-Lin Huang, Model-based articulated hand motion tracking for gesture recognition, *Image and Vision Computing*, Volume 16, Issue 2, Pages 121-134, 1998

[10] J.C. Perry, J.Rosen, S.Burns, Upper-Limb Powered Exoskeleton Design, *IEEE Transactions on Mechatronics*, Volume 12, No. 4, pp. 408-417, 2007

[11] J.Rosen, M.Brand, M.B. Fuchs, and M.Arcan. A myosignalbased powered exoskeleton system, *IEEE Transaction on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 31(3), 210– 222, 2001.

[12] Atsushi Takemoto, Ken Ichi Yano, Takanori Miyoshi, Kazuhiko Terashima, Operation assist control system of rotary crane using proposed haptic joystick as man-machine interface, *13th IEEE International Workshop on Robot and Human Interactive Communication,* Page(s): 533 - 538, 2004

[13] Toshio Tsuji, Yoshiyuki Tanaka, Tracking Control Properties of Human–Robotic Systems Based on Impedance Control, *IEEE Transaction on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol.35, No.4, 523-535, 2005

[14] Joseph J. Oresko, Zhanpeng Jin, Jung Cheng, Shimeng Huang, Yuwen Sun, Heather Duschl, Allen C. Cheng, A Wearable Smartphone-Based Platform for Real-Time Cardiovascular Disease Detecnion Via Electrocardiogram Processing, *IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE*, VOL. 14, NO. 3, 2010.

[15] Cristiano Spelta, Vincenzo Manzoni, Andrea Corti, Andrea Goggi, Sergio Matteo Savaresi, Smartphone-Based Vehicle-to-Driver/Environment Interaction System for Motorcycles, *IEEE EMBEDDED SYSTEMS LETTERS*, VOL. 2, NO. 2, JUNE 2010.

[16] Emmanouil Koukoumidis, Margaret Martonosi, Li-Shiuan Peh, Leveraging Smartphone Cameras for Collaborative Road Advisories, *IEEE TRANSACTIONS ON MOBILE COMPUTING*, VOL. 11, NO. 5, MAY 2012.

[17] Daryush D. Mehta, Matias Zanartu, Zhengran W. Feng, Harold A. Cheyne II, and Robert E. Hillman, Mobile Voice Health Monitoring Using a Wearable Accelerometer Sensor and a Smartphone Platform, *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, VOL. 59, NO. 11, 3090-3098, 2012

[18] Jeonghee Kim, Xueliang Huo, Julia Minocha, Jaimee Holbrook, Anne Laumann, Maysam Ghovanloo, Evaluation of a Smartphone Platform as a Wireless Interface Between Tongue Drive System and Electric-Powered Wheelchairs, *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, VOL. 59, NO. 6, JUNE 2012.

[19] Huy-Kyung Oh, In-Cheol Kim, Hybrid Control Architecture of the Robotic Surveillance System Using Smartphones *8th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)* , Incheon Korea, 2011

[20] Byung-Hyug Lee, Sheng-Hai An, Dong-Ryeol Shin, A Remote Control Service for OSGi-based Unmanned Vehicle using Smartphone in Ubiquitous Environment, *Third International Conference on Computational Intelligence, Communication Systems and Networks*, 2011.

[21] Jong Hyun Park, Tae Houn Song, Ji Hwan Park, Jae Wook Jeon, Usability Analysis of a PDA-based user interface for mobile robot teleoperation, *IEEE International Conference on Industrial Information (INDIN 2008)*, Daejeon, Korea, 2008

[22] Terrence Fong, Chris Provencher, Mark Micire, Myron Diftler, Gegi-nald Berka, Bill Bluethmann, David Mittman, The Human Exploration Telerobotics project: Objectives, approach, and testing, *IEEE Aerospace Conference*, 2012

[23] Saso Koceski, Natasa Koceska, Ivica Kocev, Design and evaluation of cell phone pointing interface for robot control, *International Journal of Advanced Robotic systems*, Vol9, No.135, 1-12 , 2012

[24] Amber M. Walker, David P. Miller, Tele-operated robot control using attitude aware smartphones, *7th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* , Boston, Massachusetts, USA, 2012

[25] Fausto Ferreira, Marco Bibuli, Massimo Caccia, Giorgio Bruzzone, Gabriele Bruzzone, Enhancing autonomous capabilities and human-robot interaction for unmanned sufrace vehicles, *2012 Mediterranean Conference on Control and Automation (MED)*, July 3-6, Barcelona, Spain, 2012

[26] Sung Wook Moon, Young Jin Kim, Ho Jun Myeong, Chang Soo Kim, Nam Ju Cha, Dong Hwan Kim, Implementation of Smartphone Evironment Remote Control and Monitoring System for android Operating System-based Robot Platform. International *Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Nov 23-26, 2011 in Songdo ConventiA, Incheon, Korea.

[27] Robix, http://www.robix.com/

[28] A.V.Oppenheim, R.W.Schafer, J. R.Buck, *Discrete-time signal processing*, Upper Saddle River, N.J.: Prentice Hall, 1999

[29] A.Link and H.J.Martens, Accelerometer identification using shock excitation, *Measurement*, Vol. 35, pp. 191-199, 2004.

[30] F.L.Lewis, D.M.Dawson, C.T.Abdallah, *Robot Manipulator Control: Theory and Practice*, 2nd Edition , Marcel Dekker, Inc. 2004

[31] KUKA, youbot arm, http://www.kuka-youbot.com