‘Gest-BOT’ – A highly convenient locomotive solution for the elderly and physically challenged

Rajesh Kannan Megalingam
Dept. of Electronics and Communication
Amrita Vishwa Vidyapeetham University, India
rajeshm@am.amrita.edu

Ramesh Nammily Nair, Sai Manoj Prakhya, Mithun Mohan
Dept. of Electronics and Communication
Amrita Vishwa Vidyapeetham University, India
rameshnair@ieee.org, saimanoj18@gmail.com, mithunmohan@ieee.org

Abstract – **Various research studies conducted all around the world reveal that 75% of people with debilitating physical disabilities and elderly people experience significant difficulties for their day to day locomotion. Although there are existing solutions to overcome this problem, none of them served to be customizable, economical and user friendly all at the same time. Therefore, one of the best options that could be useful for these affected class of people is a gesture-based interaction with their environment, in particular their wheelchairs. The proposed system named as ‘Gest-BOT’, uses a small camera mounted very close to the user’s hand, which tracks the small movements of their fingers to understand the direction of movement of the wheelchair. A gesture recognition system which identifies the gesture is then interfaced to the wheelchair control system in order move it to the desired location. Gest-BOT is effortless to use, customizable, economical, highly convenient and non-intrusive. The hardware implementation of Gest-BOT consists of a wheelchair, an Arduino board based ATmega microcontroller and an Intel Atom EBC-352 processor, which uses MATLAB® executables for gesture image processing. We have also calculated the response of the system under various conditions which is detailed in the paper.**

Keywords - gesture based, customizable, economical, elderly and physically challenged

I. INTRODUCTION

Our paper is conceived as an idea to ease the lives of those among us who are unfortunate enough to have lost the ability to move their limbs due to a significant amount of paralysis or due to old age – i.e. they are confined to a motorized wheelchair for locomotion. Their lives are made difficult by the fact that there is lack of an intuitive control system for their wheelchairs. The existing technologies, such as joystick-based controllers and keypad-based controllers, are not sufficiently customizable to deal with different degrees of disability. These solutions are also handicapped by the requirement that force should be applied, though however slight, in order to trigger the controls. Aiming to address these defects, we arrived at a user friendly, fully-customizable gesture-recognition-based wheelchair control system named as ‘Gest-BOT’, which is built in the most simple and economical way possible. In brief, Gest-BOT allows the seated wheelchair user to rest his hand on a smooth, frictionless surface, which has the means for capturing an IR silhouette of the hand, and control the movements of his/her wheelchair with small, pre-defined motions. These motions are customizable; it is up to the user to select the most comfortable ones depending on the degree of his/her disability. It is also a virtue of the system that even the foot can be substituted in place of the hand for users who might find that more convenient.

II. MOTIVATION

Our University is run by a NGO called the Mata Amritanandamayi Math (M.A.M). The M.A.M has around three thousand inmates of whom elders and physically challenged are about 20% of the total number. Many elders could not afford to use the electric wheelchairs due to their high cost in the market. The very few who use such wheelchairs are not comfortable with the design. A German inmate at M.A.M who is handicapped and who uses an electric wheelchair bought at Germany said that not only the wheelchair was so costly but also not very user friendly. She expressed that a gesture based wheelchair would be an ideal choice for her use. It is shown in [1] that until quite recently disabled people were socially isolated. Whether their condition was physical, emotional or mental, all met the same attitudes. They were kept off from social gatherings because they needed special attention or people to take care of them. These miserable conditions made us think of bringing out Gest-BOT, which can be used by the misfortunate people so that they can navigate easily and without external aids.

III. AVAILABLE SOLUTIONS AND SHORTCOMINGS

The current solutions as far as aids for locomotion are concerned, are custom-fitted motorized wheelchairs that have a variety of control systems based on the degree of disabilities being addressed. The most common among these are mouth-stick based control, keypad-based control, and joystick-based control and brain actuated wheelchair controls as shown from [12]. Other methods such as direct-neural control, pupil-tracking control, head-movement-based control and tongue-based control are sometimes used in severe cases of paralysis. The main shortcomings of the above systems are that they are far too expensive and suitable only for niche use. The latest rehabilitation products available in the market nowadays such

978-0-7695-4849-4/12 $26.00 © 2012 IEEE
DOI 10.1109/GHTC.2012.83

as DX-RJM-VIC-CCD finger steering control of Dynamics are solutions available wherein the user can control the motorized wheelchair using finger movements. But such systems also can be used only with one finger movement, whereas our system can be used for customizing by the users hand, fingers or even leg. The inexpensive wheelchairs available in market are not motorized and require the help of an external person for movements. This is a huge hurdle for the large majority of physically challenged and elderly people to overcome. They can exert precious little force, which may only be sufficient to just move their limbs alone. They find such solutions extremely difficult to use - movements such as pressing buttons or controlling a joystick are impossible for these people, who lack fine motor control.

IV. Our Proposed Solution

The elderly and the physically challenged often retain some imprecise motion of their fingers, though they may find it difficult to exert any sort of force or leverage. Therefore, one of the best option is a gesture-based interaction with their environment, in particular their wheelchairs. This should not be confused with the usual image associated with the words '*gesture recognition*' that comes to mind, where a person makes large motions with his hands in front of a camera. Gest-BOT uses a small camera mounted very close to the user's hand. It detects the small movements of the hand and the gesture image processing done in MATLAB® identifies which gesture has been shown by the user. The motors for the wheelchair are driven accordingly to the desired location. We have tested this on a real time wheelchair using the Intel's Embedded Computing board - Atom Board EBC 352 for image processing.

A. The heart of the system – Intel Atom EBC 352

The heart of the system as mentioned above is the Intel Atom Board EBC 352; an embedded computing board (3.5" CPU) manufactured by NexCom International Co. Ltd. The EBC 352 has an onboard Intel Atom Dual Core D525 CPU and it supports DDR3 SODIMM SDRAM, up to 2 GB. The powering up of the system is done using a DC-DC SMPS (ATX base, 24 pin, 240 Watts) which can supply 12 V DC input to the power pins of the board. We have converted the '.m' format MATLAB® files into '.exe' executable files that can be run of the Atom board even without the installation of MATLAB® in it. The conversion can be performed using the inbuilt commands on the MATLAB® installed PC. A 140 GB hard disk is interfaced with the Atom board so that Windows 7 operating system is installed onto it. The power requirements for the hard disk are similar to as powering up a normal SATA hard disk on any computer. A 5V DC and a 12 V DC which comes out from a normal SMPS will do the job of powering up the SATA hard disk.

V. Related Works

Ref. [3] discusses about a Grey-Fuzzy Decision-Making (GFD) algorithm based fuzzy logic theory to avoid potential accidents in wheelchairs. It discusses about an algorithm to predict the slippage of electric wheelchairs. As our wheelchair is basically used for indoor home navigation, we do not need to have precautionary measures for slippage, but we plan to implement collision avoidance which is discussed later in the paper. Ref. [4] shows a gesture based wheelchair, which makes use of the ultrasonic sensors to avoid obstacle avoidance in crowd. We are also planning to use ultrasonic sensors on our wheelchair which can serve the purpose of obstacle avoidance. But the gestures mentioned in this paper are made using head movements and hence it can be tedious task for the user. Ref. [5] explains the defects of ultrasonic sensors and its usage defects in crowd. They also discuss about the specification constraints to be followed in choosing a perfect ultrasonic sensors for our application. Ref. [6] discusses a navigation system which makes use of sonar maps to get location identity. It makes use of the conventional joystick control wheelchairs. As the paper discusses about using the conventional joystick bases controls, it cannot help for the worst class of affected physically challenged people. Findings from [7] reveals that gesture based navigation system that are extremely useful for the users who have restricted limb movements caused by some diseases such as Parkinson's disease and quadriplegics. The authors of [7] also explain about the merits of using a gesture based interaction in the wheelchair control system for the quadriplegics. The authors of [8] explain about an obstacle avoidance wheelchair that works based on the face of the pedestrian. But the main defect of [8] is that this system does not work when the pedestrian walks in the same direction of the wheelchair and the person riding the wheelchair does not notice the pedestrian. They haven't mentioned any solutions for the above mentioned problems also. Ref. [9] discusses about a gesture based and vision driven wheelchair for the physically challenged. Mapping the eye movements onto a display to track the specific tasks are used here but this result in huge amount of stress on the user.

VI. System Description

A. Gesture Capture Module (GCM)

Capturing the image of the gesture performed by the user of the wheelchair is done using an IR-camera and MATLAB®. A normal inexpensive USB web-camera is not sensitive to IR radiations. We have modified a normal web-camera to become sensitive to infrared radiation by removing the IR filter from it. We are capturing the silhouettes of the gestures lit from the back using the IR-sensitive webcam mounted in front. These images are fed as inputs to the gesture recognition module via the camera's USB interface. MATLAB® can directly acquire images from the web-camera using its inbuilt functions. The user's hand is placed on the palm-rest of the wheelchair. The camera for capturing the gesture is mounted just above the hand and focussed downwards; we require backlighting of the surface the user has placed his/her hands on.

1) System Customization: The user has to predefine the gesture templates that he/she is going to use for any of the directional movements viz., Forward, Reverse, Turn Left, Turn Right or Brake. The user may use only one finger or all the fingers or even the leg for each of the gesture. This is a simple exercise in which the system requires at least 4 set of gestures to be captured initially and stored as image templates before the user can start using the wheelchair. This way each user can define his/her own gesture image templates according to their convenience. These set of images are called predefined gesture templates and the complete process is defined as System Customization. Figure 1 shows the block diagram of the complete system and the various modules that are used.

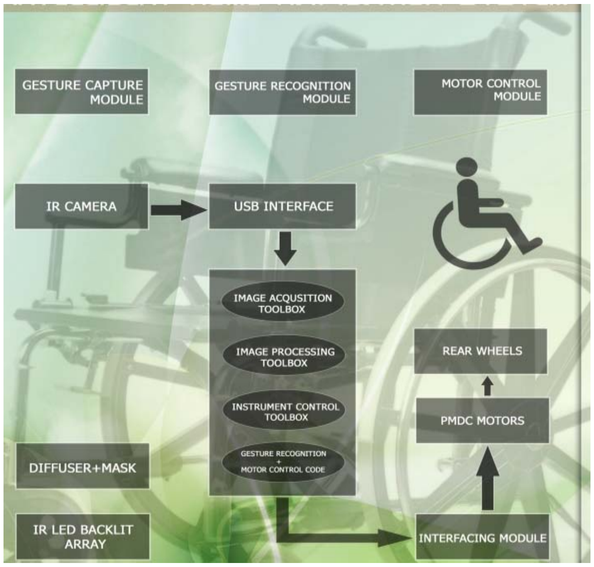

Figure 1. System Block Diagram

B. Gesture Recognition Module (GRM)

When the system is in use, the captured gesture images have to be processed to identify the correct gesture. These images may match with any of the predefined images. This technique of template matching is used to identify the correct gestures. The captured images are pre-processed, converting them into black and white images. These binary images are correlated in real time with all the previously stored templates. The template with maximum correlation is identified. If the value of this correlation is greater than or equal to a pre-defined threshold, a successful validation occurs and messages are sent to the successive module. The experimental results to understand the correlation value is described in the section called Experimental Results. Figure 2 shows the set of gestures that are produced for various movements like forward, reverse etc. The first figure under each head shows the hand being placed to produce the gesture for the respective movement and the next figure shows its conversion into black and white image.

(a)

(b)

(c)

(d)

(e)

Figure 2. Suggested Gestures; (a) Brake, (b) Forward, (c) Reverse, (d) Turn Left, and (e) Turn Right.

A set of four images for each of the five gestures produces a total of twenty gestures. However, 2D cross-correlation done with twenty images of resolution 320 x 240 pixels is a time-consuming process. Hence, every sixth frame of the raw image data that comes at a rate of 15 frames/second from the camera is picked for the recognition purpose. During the time when the system is not in use, the correlation procedure has time to complete. Once this correlation process is completed, the current frame captured is identified and recognized as a gesture only if it has a correlation value greater than or equal to 0.65 with one of the templates. Otherwise, it is identified as an invalid one and the default action of 'Brake' occurs. A Graphical User Interface (GUI) has been designed in MATLAB® for ease of use during the customization phase and for debugging further operation. It provides the means for creating new templates, watching a live preview from the camera, testing a snapshot and starting or stopping the control module.

C. Interfacing Module (IM)

The interfacing module refers to the interface between the GRM with Atom board and the motors of the wheelchair. The Arduino based microcontroller board is used to drive the wheelchair motors. The interfacing module can be implemented by driving the Arduino board through the USB port of computer. Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The Arduino uses ATmega168 microcontroller as its core. It has 14 digital I/O pins, 6 analogue inputs, 16 MHz crystal oscillator, a USB connection and a power jack. It can transmit and receive TTL serial data. It also has SPI and I^2C interfaces. The Arduino board can be powered up using USB in standalone mode. We have used the Arduino Duemelinova board, as it can be interfaced with the Atom board using the USB port and it is also an open source electronics prototyping platform.

D. Motor Control Module (MCM)

This module is used to control the motors of the real time wheelchair. Two motors connected to the rear wheels of the wheelchair have to be controlled to define its motion. We have used Permanent magnet DC motors (PMDC) for the movements of the wheelchair. The PMDC motors can be controlled using Pulse Width Modulation (PWM) techniques that can be generated using the Arduino microcontroller. The power wheelchair can be directionally controlled using suitable motor driver. We have used a motor driver in between the Arduino board and the PMDC motors. We used Super Hercules 9V - 24 V, 15 A motor drivers to control the motors. Super Hercules 9V – 24V motor drivers can take up to 30A peak current load and can be operated up to 10 KHz PWM. It also gives out false diagnostic outputs and pre settable overload protection limit. The direction of the movement can be controlled using a separate pin on the motor driver which is used for this specific purpose as shown in [11]. Hence the wheelchair can also be made to move in the reverse direction with the help of the direction pin on the motor controller. The various commands with values that are written to the microcontroller on the Arduino board such as analogWrite(0) for 0% duty cycle, analogWrite(127) for 50% duty cycle etc. to generate the desired PWM outputs. The maximum value that can be written is 255 as the microcontroller used has only 8 bit digital output data.

VII. REAL TIME HARDWARE SETUP

Fig. 3 shows the final setup of the real time gesture based wheelchair named as 'Gest-BOT'. By the word 'real time' we mean that a person can actually sit on our wheelchair and move to various directions accordingly. The gestures of the hand are captured by placing the hand in the gesture capture module (the brown colored box fixed adjacent to the hand rest of the wheelchair). It is to be noted that there is a camera placed inside the brown box and the illumination with IR Led's as discussed earlier is also inside the box. The GRM is taken care by the Intel Atom EBC 352 processor which is kept beneath the wheelchair. There are also 24 V, 26 Ah batteries that are kept under the wheelchair for powering up the motor drivers and for successfully running the motors. The cylindrical shaped motors are fixed next to the wheels of the wheelchair. The Arduino board which takes care of the IM is also kept next to the Intel Atom processor inside the box which is labeled in the figure below. So to summarize, 'Gest-BOT' – the real time wheelchair developed consists of a basic model of a wheelchair, Batteries for powering up, PMDC motors, GCM including camera, GRM which is taken care by the Intel Atom board, IM by the Arduino board and the MCM with the help of Hercules Motor drivers and motors.

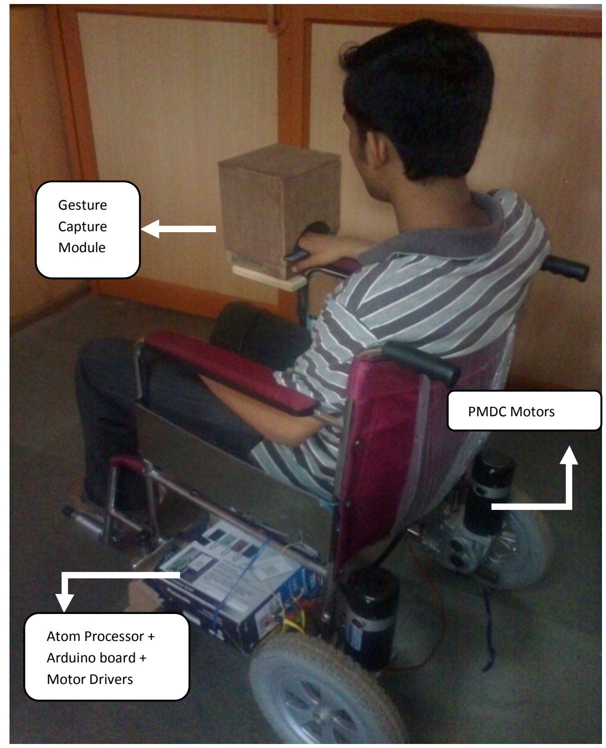

Figure 3. Final Setup of 'Gest-Bot'

VIII. CHARACTERISTIC BEHAVIOUR AND PARAMETERS

One would expect that as soon as a valid gesture is performed in front of the camera in the GCM, the GRM would immediately recognize it and the wheels of the wheelchair would instantly turn in the expected direction, or not at all for an invalid gesture. However, this is only the ideal case. There are some deviations in the real case. Those are discussed below:

A. Deviation

There are several factors that can cause deviations from the expected behavior. The accuracy of gesture detection is governed by some external factors, despite our best efforts to make the capture module insensitive to these. The customized templates defined by the user are a source of weakness – if the gestures for each command are not sufficiently differentiated from the others, it can lead to inaccuracy in detection. If the user significantly changes his hand position after recording the template, there may not be in sufficient correlation to recognize a gesture, even if the configuration itself is correct. The threshold used to reject invalid gestures may have to be tweaked for better accuracy. The wheels and the motors attached to them may also require some time for spinning up from rest, or coming to rest from active rotation, or reversing direction.

B. Parameters

The different parameters that govern the behavior of the system are the gesture rejection threshold G_{Th}, the frame capture interval T_f, the number of templates per command N_T and the resolution of the captured gesture w_{res} x h_{res}. The gesture rejection threshold is the minimum value of the 2D correlation that is required for a template match to pass muster and be recognized as a valid gesture command. Through experimentation, a G_{Th} value of 0.65 was found and fixed. Increasing the value of G_{Th} makes the system recognize gestures that conform more strictly to the recorded template. This means that there is more difficulty in recreating a valid gesture. Decreasing the value, on the other hand, makes it easier to validate a gesture, but it is more likely that the system will make mistakes in rejecting stray gestures. The frame capture interval T_f is fixed on the basis of how long the system takes to correlate the templates. With the present hardware, we could capture up to 15 frames per second. However, we correlate every 6^{th} frame only. Therefore, the value of T_f is set at 400 ms. The number of templates per command N_T is set at 4. When the value of N_T is increased, there is more chance of successfully recognizing a gesture, but the processing time increases linearly, and vice versa. The resolution of the captured gesture w_{res} x h_{res} is set at 320 x 240 pixels.

C. Jerk Avoidance

The control signals which include PWM and direction signals are given by Arduino board to Hercules motor driver. The Hercules motor driver acts as a switch between the input 5 volt signals and required 24 volt output signals. The wheelchair moves at a considerably good speed when it is given 15 volts. Thus to prevent the jerk or a sudden unwanted vibration when the wheelchair starts or stops, the required 15 volts is reached by incrementing PWM duty cycle in linear steps. Even then, there were jerks when the wheelchair had to move from one direction to another, say from right to left. This was solved by making the wheelchair to have an intermediate state of brake before transition from one direction to another. Thus whenever there is a sudden change in the direction, the wheelchair slows down and gains the required amount of speed in expected direction. Thus the problem of jerk was reduced considerably in Gest-BOT.

IX. Experimental Results

We conducted various experiments to determine the best fitting correlation value for our image comparison algorithm used in the GRM. For six different values of correlation values the response time was calculated. The six values taken are 0.55, 0.6, 0.65, 0.7, 0.75 and 0.8. For each of these values the response time for each of the gesture was different. The graph shown in Figure 4 shows the time taken on the Y axis and the various correlation values on the X axis. The best response was noted at a correlation value of 0.65, so we finalised 0.65 as the correlation value. By the word best response, we mean that the system responded more quickly in the particular direction when a gesture was given. As shown in Figure 4, for the correlation value of 0.65, we noted that our Gest-BOT responded in less than 0.59 seconds. The system took more time to respond when the other correlation values were given. Figure 4 shows the correlation value vs. time taken for producing the reverse gesture and Figure 5 shows the same for the forward gesture. For both the cases we could see that 0.65 was the best correlation value that could be used. Similar patterns of graph followed for other gestures also.

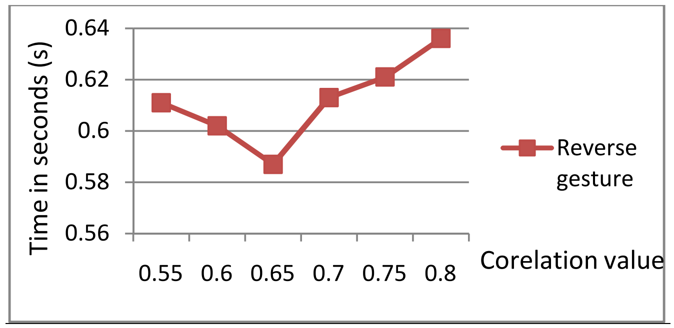

Figure 4. Correlation value vs. Time taken in (sec) for reverse gesture

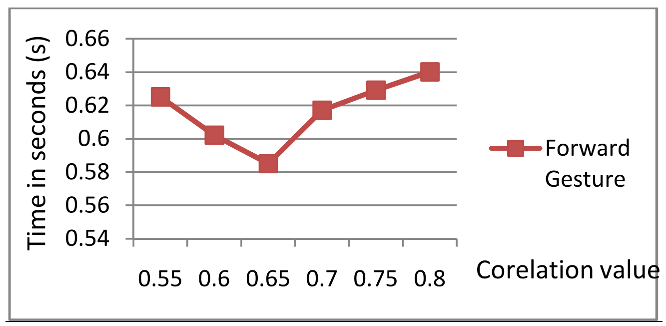

Figure 5. Correlation value vs. Time taken in (sec) for forward gesture

Table I and Figure 6, denotes the response time for various gesture changes (as shown in figure) under two situations – blue bar indicating the case when there are no jerks on the wheelchair movement and the second case indicating the case where there were jerks during the wheelchair movement. The time was plotted in the Y axis and the various movements were plotted in the X axis.

TABLE I
RESPONSE TIME (IN SECONDS) FOR VARIOUS GESTURE CHANGES

Gesture Change	Response time without jerk (sec)	Response time with jerks (sec)
Forward to brake	1.27	1.22
Reverse to brake	1.23	1.41
Right to brake	1.25	1.33
Left to brake	1.16	1.82

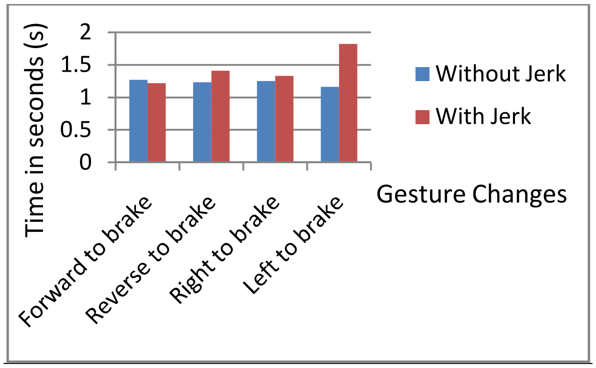

Fig. 6 – Time taken for various gesture changes with and without jerks

Table II and Figure 7, denotes the response time for various gesture changes (as shown in figure) under two situations – blue bar indicating the case when the gesture image processing was done using Intel Core 2 Duo processor (on a laptop) and the second case indicating the case when the gesture image processing was done using Intel Atom EBC 352 processor. The time was plotted in the Y axis and the various movements were plotted in the X axis.

TABLE II
RESPONSE TIME (IN SECONDS) FOR JERKS AND WITHOUT JERK MOVEMENTS

Gesture Change	Response Time (in sec) in Intel Core 2 Duo	Response Time (in sec) in Intel Atom EBC 352
None to Forward	1.22	0.8
Forward to Left	0.88	0.6
Left to Right	1.33	0.7
Right to Left	1.88	1.1
Left to None	1.82	1.06

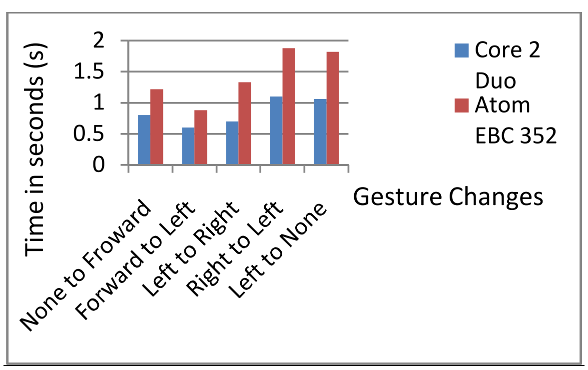

Fig. 7 – Time taken for various gesture changes with Intel Core 2 Duo and Atom processors

X. CONCLUSION

This paper discusses on development of 'Gest-BOT' - effortless to use, customizable, economical, highly convenient and non-intrusive gesture-based wheelchair control system for elderly and physically challenged. These individuals can make use of small gestures of their fingers, hands or feet in order to control the motion of their wheelchairs. Four modules are serially connected in order to achieve this – Gesture Capture Module, Gesture Recognition Module, Interfacing Module and Motor Control Module. Nowadays the electronic wheelchairs are mainly controlled using joysticks or keypads. But this gesture based system is economical, effortless to use and user friendly when compared with the existing systems. Thus our system will definitely help all the misfortunate physically challenged and elderly people. This wheelchair can be used by a physically challenged or a handicapped person who require external aid for their day today locomotion. By the use of our wheelchair, they no longer require external aid for their movements inside their house or surroundings.

ACKNOWLEDGMENT

The authors wish to thank Amrita Vishwa Vidyapeetham University, Amritapuri, India for their infrastructure support and IEEE Foundation, NJ, USA which is funding this project.

REFERENCES

[1]. Special Needs and Daily living – Special needs Solutions for all.

[2]. Ase brandt, Susanne iwarsson and Agneta sta, "Older People's Use of Powered Wheelchairs for Activity And Participation", Taylor and Francis health Sciences

[3]. Ren C. Luo, Tse Min Chen, Chi-Yang Hu, and Zu Hung Hsiao, 'Adaptive Intelligent Assistance Control of Electrical Wheelchairs by Grey-Fuzzy Decision-Making Algorithm' , Proceedings of the 1999 IEEE International Conference on Robotics & Automation Detroit, Michigan May 1999.

[4]. Yoshinori Kunotl, Teruhisa Murashimat , Nobutaka Shimadat and Yoshiaki Shirait 'Interactive Gesture Interface for Intelligent Wheelchairs'

[5]. Donald P. Massa, 'Choosing an Ultrasonic Sensor for Proximity or Distance Measurement Part 1: Acoustic Considerations'

[6]. S. Fioretti, T. Leo, and S. Longhi, 'A Navigation System for Increasing the Autonomy and the Security of Powered Wheelchairs', IEEE Transactions On Rehabilitation Engineering, Vol. 8, No. 4, December 2000

[7]. Pei Jia, Huosheng H Hu, Tao Lu Kui Yuan, 'Head gesture recognition for hands-free control of an intelligent wheelchair',

[8]. Yoshifumi Murakamit , Yoshinori Kuno, Nobutaka Shimadat and Yoshia, 'Collision Avoidance by Observing Pedestrians Faces for Intelligent Wheelchairs', Proceedings of *the 2001* IEFWRSJ International Confermm on Intelligent Robots Mwi,Hawaii, USA, *Oct. 29*-NOV0.3,2009

[9]. Rajat Garg, N.Shriram, Vikrant Gupta, and Vineet Agrawal, 'A Smart Mobility Solution for Physically Challenged', ICVES 2009, 978-1-4244-5441-9/09/$26.00

[10]. User Manual – Intel Atom Processor EBC 352

[11]. Product Manual of Super Hercules 9V – 24V, 15 A motor drivers from Nex Robotics

[12]. Arduino Board and related downloads – www.arduino.cc Sakiko Tashiro and Toshiyuki Murokami, "Step passage control of a power assisted wheelchair for a caregiver", IEEE Transactions on Industrial Electronics, Vol. 25, No. 4, April 2008