الجمهورية الجزائرية الديمقراطية الشعبية
RépubliqueAlgérienneDémocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

Université Mohamed Khider – Biskra

Faculté des Sciences et de la technologie

Département : Génie Mécanique

Ref :……………

جامعة محمد خيضر بسكرة
كليةالعلوم والتكنولوجيا
قسم: الهندسة الكهربائية
المرجع:……….

Thèse présentée en vue de l'obtention

Du diplôme de

Doctorat LMD en : Automatique

Option : Estimation et contrôle des systèmes

Multimodal Control of an Intelligent Wheelchair

(Commande Multimodale d'un Fauteuil Roulant Intelligent)

Présentée par :

HASSANI Rafia

Soutenue publiquement le : 14 / 10 / 2020

Devant les membres de Jury :

Dr BAHRI Mebarik	**Professeur**	**Président**	**Université de Biskra**
Dr BOUMEHRAZ Mohammed	**Professeur**	**Rapporteur**	**Université de Biskra**
Dr TERKI Nedjiba	**Professeur**	**Examinatrice**	**Université de Biskra**
Dr HEFAIFA Ahmed	**Professeur**	**Examinateur**	**Université de Djelfa**

Acknowledgements

First of all, I wish to express my sincere gratitude to my teacher and my advisor, Mr. BOUMEHRAZ Mohamed, Professor of University Mohammed Khider Biskra. I am highly indebted and appreciated him for giving me the chance , advices, support, and encouragement throughout the research period.

I would like to thank my jury members, namely Mr. BAHRI Mebarik, Professor of University Mohammed Khider Biskra, Mrs. TERKI Nadjiba , Professor of University Mohammed Khider Biskra and Mr. HAFAIFA Ahmed, Professor of University Ziane Achour Djelfa. Their time and energy are gratefully acknowledged. I want to appreciate very much for their detailed reviews and insightful comments.

Moreover, I would especially like to thank all the Electrical Engineering department staff, for their professional treatment of me as a colleague.

I would thank my family for their patience and encouragement during my study.

I would like to thank all my colleagues and friends whose company and encouragement helped me a lot to work hard.

Last, but not the least, I thank ALLAH, for giving me the strength during the course of this research work.

Dedication

To my Family,

to my dear husband

and my lovely son and daughter

Abstract

An intelligent wheelchair (IW) is a standard electric wheelchair (EW) to which a computer and a collection of sensors have been added, giving the user, who cannot use the usual joystick, the ability to control it without touching any physical device. In the recent years, several research projects around the world have been focused on the development of IW prototypes. However, the adaptation of their user interface to the patient's abilities is often a neglected research topic. The majority of the interfaces are adapted to a specific user group.

This thesis presents the design and implementation of a multimodal control system for IW to assist people with different levels of disabilities. The system is composed of an EW equipped with necessary electronics (microcontrollers, sensors, laptop, microphone and camera) to ensure safe mobility and ease of operation.

In order to control the movement of the wheelchair, firstly, one of the control modes (manual, vision based method, sensors based method, or speech based method) is selected by the user according to his/her abilities. Then, depending on the controlling command (Forward, Backward, Right, Left, or Stop) the microcontroller generates specific signals to emulate the joystick for monitoring the control board of the wheelchair. The principal advantage of the proposed system is that the switching between the control modes is soft, straightforward and transparent to the user. The experimental tests applied to this wheelchair in real environment, proved the feasibility and efficiency of our multimodal system.

Keywords: Intelligent Wheelchair, Multimodal control, Microcontroller, Sensor.

Résumé

Un fauteuil roulant intelligent est un fauteuil roulant électrique standard auquel un ordinateur et un ensemble de capteurs ont été ajoutés, donnant à l'utilisateur, incapable d'utiliser le joystick conventionnel, la possibilité de le contrôler sans toucher aucun dispositif physique.

Au cours de ces dernières années, plusieurs projets de recherche dans le monde se sont concentrés sur le développement de prototypes de fauteuils roulants intelligents, mais l'adaptation de leur interface d'utilisateur avec la capacité du patient est souvent un sujet de recherche négligé. La majorité des interfaces sont adaptées à un seul utilisateur ou à un groupe spécifique des utilisateurs.

Cette thèse présente la conception et l'implémentation d'un système de commande multimodale d'un fauteuil roulant intelligent afin d'aider les personnes, ayant différents niveaux de handicap, de piloter sans difficultés le fauteuil roulant. Le système est composé d'un fauteuil roulant électrique équipé par des dispositifs électroniques (microcontrôleurs, capteurs, ordinateur portable, microphone et caméra) pour assurer la mobilité en toute sécurité et faciliter l'utilisation.

Afin de contrôler le mouvement du fauteuil roulant, premièrement, l'un des modes de contrôle (command manuelle, commande basée sur la vision, commande basé sur les capteurs, ou commande vocale) est sélectionné par l'utilisateur selon ses capacités. Ensuite, selon la commande désirée (avant, arrière, droite, gauche ou arrêt), le microcontrôleur génère des signaux similaires aux signaux générés par le joystick pour contrôler le circuit de commande du fauteuil roulant. L'avantage principal de système proposé est que la commutation entre le mode de contrôle choisi et le mode conventionnel est douce, directe et transparente pour l'utilisateur. Les tests expérimentaux appliqués à ce fauteuil roulant en environnement réel ont prouvé la faisabilité et l'efficacité de notre système de commande multimodale.

Mots-clés: fauteuil roulant intelligent, commande multimodale, microcontrôleur, capteur.

ملخص

الكرسي المتحرك الذكي هو كرسي متحرك كهربائي أضيف إليه جهاز الحاسوب و مجموعة من أجهزة الاستشعار، مما يسمح للمستعمل الغير قادر على استخدام عصا التحكم العادية القدرة على التحكم في الكرسي المتحرك دون الحاجة إلى لمس أي جهاز. ركزت العديد من المشاريع البحثية في السنوات الأخيرة عبر كافة أنحاء العالم على تطوير نماذج للكرسي المتحرك. غير أن تكيف واجهة المستخدم لهذه الكراسي المتحركة مع قدرات المريض هي في الغالب موضوع بحث مهمل. فغالبية هذه الواجهات مكيفة لمستخدم واحد أو لمجموعة محددة من المستخدمين.

تقدم في هذه الرسالة تصميم و تطبيق نظام تحكم متعدد الوسائط للكرسي المتحرك الذكي لمساعدة الأشخاص ذوي الإعاقة بمستويات مختلفة. يتكون النظام من كرسي متحرك كهربائي مزود بالمعدات الاليكترونية اللازمة (المتحكمات الدقيقة، أجهزة الاستشعار، حاسوب محمول، ميكروفون، و كاميرا) لضمان التنقل الآمن و سهولة الاستعمال.

من أجل التحكم في حركة الكرسي المتحرك، يقوم المستخدم أولا باختيار طريقة واحدة من الطرق المتوفرة للتحكم في الواجهة وفقا لقدراته (الطريقة اليدوية، طرقة تعتمد على الرؤية، طريقة تعتمد على أجهزة الاستشعار، أو الطريقة التي تعتمد على الكلام). ثم بناءا على الأمر المراد تطبيقه (أمام، وراء، يمين، يسار، أو التوقف)، يقوم المتحكم الدقيق بتوليد إشارات مماثلة للإشارات التي تولدها عصا التحكم وذلك للسيطرة على نظام التحكم للكرسي المتحرك. تتمثل الميزة الرئيسية للنهج المقترح في أن التبديل بين أوضاع التحكم يكون بسيطًا وواضحًا وشفافًا للمستخدم. أثبتت الاختبارات التجريبية المطبقة على هذا الكرسي المتحرك في المحيط الواقعي جدوى وكفاءة نظامنا المتعدد الوسائط.

الكلمات المفتاحية: كرسي المتحرك الذكي، تحكم متعدد الوسائط، المتحكم الدقيق، جهاز الاستشعار.

SCIENTIFIC PRODUCTIONS

Journal Papers:

- Hassani, R., Boumehraz, M., Hamzi, M., & Habba,Z. (2018). "Gyro-Accelerometer based Control of an Intelligent Wheelchair", *Journal of Applied Engineering Science & Technology*, 4(1), 101-107.

- Hassani, R., Boumehraz, &M., Hamzi, M., "***Implementation of Wheelchair Controller using mouth and tongue gesture***", Accepted in *IJEECS* Indonesian *Journal of Electrical Engineering and Computer Science.*

Communications:

- Hassani, R., Boumehraz, & Habba, Z. (2015). "An Intelligent Wheelchair Control System based on Head Gesture Recognition", *International Electrical and Computer Engineering Conference IECEC2015*, Setif, Algeria.
- Hassani, R., Boumehraz, M., & Hamzi, M. (2018). "Head Gesture Based Wheelchair Control", *Second International Conference on Electrical Engineering ICEEB'18*, Biskra, Algeria.
- Hamzi, M., Boumehraz, M., & Hassani, R. (2018). "Hand Gesture Based Wheelchair Control", *Second International Conference on Electrical Engineering ICEEB'18*, Biskra, Algeria.

ABBREVIATIONS

EEG	**E**lectro**E**ncephalo**G**raphy
EMG	**E**lectro**M**yo**G**raphy
EOG	**E**lectro**O**culo**G**raphy
HCI	**H**uman **C**omputer **I**nterface
HMI	**H**uman **M**achin **I**nterface
IMU	**I**nertial **M**easurement **U**nit
IW	**I**ntelligent **W**heelchair
MEMS	**M**icro-**E**lectro-**M**echanical **S**ystems
NAN	**N**arrow **A**rea **N**avigation
MWSR	**M**icrosoft **W**indows **S**peech **R**ecognition.
SDK	**S**oftware **D**evelopment **K**it
VAHM	French acronym for Autonomous Vehicle for people with Motor Disabilities (**V**éhicule **A**utonome pour **H**andicapé **M**oteur)
WAN	**W**ide **A**rea **N**avigation

CONTENTS

Chapter 3:

VISION-BASED HUMAN MACHINE INTERFACE SYSTEM

Chapter 4:

SENSOR-BASED HUMAN MACHINE INTERFACE SYSTEM

Chapter 5:

SYSTEM IMPLEMENTATION

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

According to data provided by Word health organization, over a billion people are estimated to live with some form of disability. This corresponds to about 15% of the word population. Between 110 million (2.2%) and 190 million (3.8%) people adult has significant difficulties in functioning. Furthermore, the rate of disability are increasing due to ageing populations and an increase in chronic health conditions such as diabetes, cardiovascular disease, cancer and mental health disorders. Traffic and work accidents, war and land mines are other factors which contribute to the increase of people with mobility difficulties [78].

Due to this high proportion, there is growing demand for developing technologies that can aid this population group from international health care organization, universities and companies interested in developing and adapting new products. Wheelchairs are important locomotion devices for these individual. However, traditional wheelchairs which are controlled by users via joystick, buttons and levers, cannot satisfy the needs of disabled users who have lost the ability to use their arms such as quadriplegia, amputee's hands and paralyzed patients. This segment of peoples needs to use special control systems depending on their abilities. In order to solve this problem, different methods and techniques have been developed to create intelligent wheelchairs (IWs) over the last years [2, 79].

Due to the low cost and widespread availability of commercial voice recognition hardware and software, voice recognition has often been used for IW [4, 6, 14, 70, and 74].

Vision based methods are another technique which uses the recognition of head gesture [33, 38], hand gesture [83], head tilt and mouth shape [40], facial expressions [68] and eye gaze tracking [62] as the main input to guide the wheelchair.

Recently, several approaches of operating an IW using advanced sensors as an input method have been proposed in many projects, such as, Micro-Electro-Mechanical Systems (MEMS) sensors [31,34, 35], sensors that capture the user's bio-signal (Electromyography [15, 32, 67, and 79], Electro-oculography [17, 80]. and Electro-encephalography [16, 43]), Tongue Operated Magnetic sensor [42], and pressure sensors [39, 81].

Most of the proposed IW projects use two basic techniques:

- Auto-navigation for automatic obstacle avoidance
- Convenient interfaces that allow disabled users to control the IW themselves using their limited physical abilities.

However, the majority of the interfaces were adaptable to a single user or to a specific user group.

This thesis presents a multimodal control system of IW. The area of multimodal control researches is not new, but in this work we tried to cover the needs of the most disabled people whatever the level of their disability by developing and designing an IW prototype with multimodal interface which combines all the possible input methods and using necessary electronics like (Microcontroller and sensors) to ensure safe mobility and ease of operation.

The remainder of this document is organized as follows.

Chapter 1: Introduces the state of the art of the IW projects, their operating modes, some low cost input devices that can be used in IW system, and the most popular mode of human machine interface (HMI).

Chapter 2: In this chapter, HMI system based on speech recognition is presented with two different ways, the first way uses Microsoft Windows speech Recognition and Microsoft visual studio C#. The other way is using an android application in Smartphone which uses Google Speech to Text to recognize and process human speech.

Chapter 3: Two vision based interface systems for controlling the wheelchair are proposed. The first interface uses the head gesture and the second uses the mouth gesture as input methods to control the movement of the wheelchair.

Chapter 4: This chapter presents an interface based on IMU sensor. The system uses the head movement and orientation to control the intelligent wheelchair. The user's head angles around the X and Y axis are interpreted as a wheelchair movement commands in the forward, backward, left and right directions

Throughout the last three chapters, the system is tested many times by two persons. The exprimental results show that the proposed interfaces can achieve the purpose of controlling the intelligent wheelchair.

Chapter 5: This chapter is divided into two parts: the first part shows the concept, design and implementation of the platform. As well as different hardware and software used for the development of our IW. In the second part, most of the proposed interfaces were combined together into a multimodal interface, implemented and tested in real environment and compared with the manual mode to evaluate their performance.

We end this manuscript with a general conclusion summarizing what has been done and the prospects for this work.

CHAPTER 1

STATE OF THE ART ABOUT INTELLIGENT WHEELCHAIRS

1.1. Introduction:

Nowadays, there is a significant increase in the number of older persons in all countries of the world. This segment of society, as well as patients with quadriplegia and amputee's hands, lost the ability to use their arms to control traditional wheelchair which is controlled by Joystick. They need to use special control systems to control electric wheelchairs.

In order to provide a better quality of life for people with this kind of disability, various alternative control methods and techniques have been developed to create intelligent wheelchairs. An Intelligent Wheelchair (IW) is a standard electric wheelchair with an embedded computer and sensors, giving it certain intelligence.

This chapter presents an overview of the most popular IW projects, a short description of the used sensors, each research focus, as well as different approaches used by researchers to solve common problems.

The outline of this chapter is as follow: at first, we briefly introduce some IW projects and the necessary requirements for such project. The second part contains an overview of some low cost input devices that can be used in IW systems. Then we present the operating modes of IW projects. Finally, the most popular mode of human machine interface is covered at the end of the chapter.

1.2. Overview of intelligent wheelchair projects

In this part we will present an overview of some IW projects, a brief description of its functionalities as well as the types of sensors used by each research group.

- **Wheelchair by Madarasz** (Arizona state university, USA, 1986)

It was basically a self-navigating wheelchair for disabled people. It was equipped with an on-board computer, a digital camera, and a scanning ultrasonic range finder and was able to navigate in corridor environments [47].

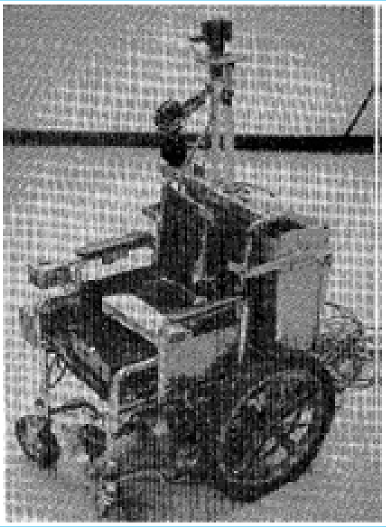


Figure 1.1: Prototype developed by Madarasz

- **The Smart Wheelchair** (University of Edinburgh, Scotland, 1995)

It was built for children that have difficulties to operate and drive standard wheelchairs (Figure 1.2). The system has bump sensors to sense obstacles and an implemented a line following algorithm, for driving through doors and between rooms [56].

Figure 1.2: Smart Wheelchair Prototype.

- **RobChair** (University of Coimbra, Portugal, 1998)

RobChair is an electric wheelchair equipped with five wheels (Figure 1.3). It could be operated by voice, keyboard, and/or by an analog joystick. It had 12 infrared sensors and four ultrasonic sensors. One tactile bumper was located in the front of the vehicle. It used potential field methods for obstacle avoidance [59].

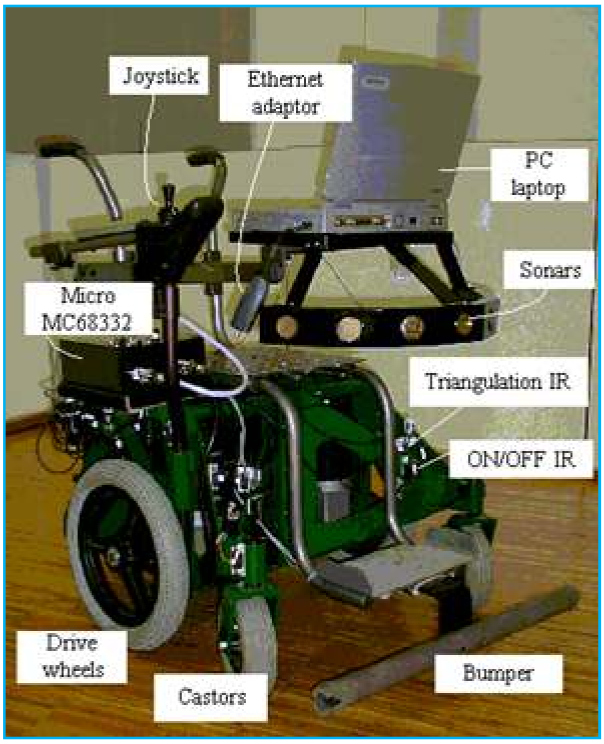


Figure 1.3: RobChair Prototype.

- **NavChair** (University of Michigan, USA, 1999)

NavChair was developed for people that suffered from different sorts of impairments such as bad vision (Figure 1.4). It was equipped with ultrasonic sensors and was able to avoid obstacle, follow wall and pass through doors. A voice control module was also adapted to this wheelchair [45].

Figure 1.4: NavChair Project.

- **FRIEND** (University of Bremen, Germany, 2001)

The project FRIEND consists of an electric wheelchair with the robot arm MANUS [51]. Both devices are controlled by speech recognition using a voice input module installed in a computer connected to the system (Figure 1.5).

Figure 1.5: FRIEND wheelchair project

- **MAid** (University of Ulm, Germany, 2001)

MAid (Mobility Aid for Elderly and Disabled People) is equipped with a modular sonar system, two infrared sensor, and a laser range finder. It can do narrow area navigation (NAN) and wide area navigation (WAN). WAN mode is used when the wheelchair is located in dynamically changing environments, such as crowded places [60].

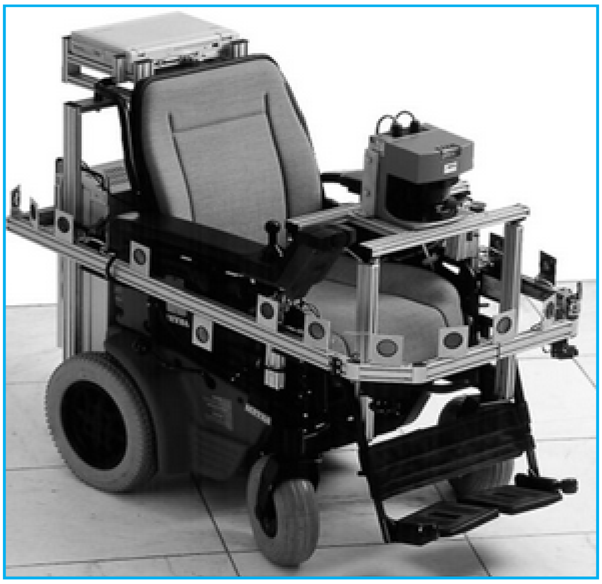


Figure 1.6: MAid prototype.

- **SmartChair** (GRASP Laboratory, University of Pennsylvania, USA, 2002)

The wheelchair is equipped with wheel encoders, an omni directional camera, IR-sensors, a laser range finder, and an interaction camera. The research focus is on vision-based human interaction [63]. It has six behaviors: hallway navigation, three-point turn (reversing and turning), obstacle avoidance, navigate through a doorway, turn while avoiding obstacle, and go to a specific goal.

Figure 1.7: SmartChair prototype.

- **VAHM** (University of Metz, France, 1993, 2004)

First VAHM was built on top of a mobile robot base. Three-level control architecture provided autonomous navigation (based on internal map) or two semi-autonomous behaviors (wall following, obstacle avoidance) and mode decisions are made manually. Second and third VAHM (Figure 1.8) based on modified power wheelchair. Use same three-level control architecture, mapping schemes, and IR beacons. VAHM provides autonomous navigation and semi-autonomous behaviors and mode decisions are made manually [9, 61].

Figure 1.8: VAHM2 (to the left) VAHM3 (to the right)

- **Rolland** (University of Bremen, Germany, 1997, 2005)

There were three different prototypes throughout the whole development period. Rolland I is based on sonar, IR and bump sensors to avoid collisions. The system is able to autonomously navigate between points on its map. Rolland II uses sonar and dead reckoning to trigger mode changes (Figure 1.9). User teaches trajectory using turn-in-place and wall-following behaviors, and trajectory can then be repeated. Rolland III used wheel encoders and omni directional camera system for solving fundamental tasks like self-localization, mapping and autonomous navigation [44, 50].

Figure 1.9: Rolland II (to the left) Rolland III (to the right)

- **The Walking Wheelchair** (University of Castilla-La Mancha, Spain, 2006)

The Walking Wheelchair is equipped with four wheels and four legs (Figure 1.10). The wheels are located at the end of each leg. The legs make it possible for the vehicle to climb stairs. The project focuses on the mechanical construction [54].

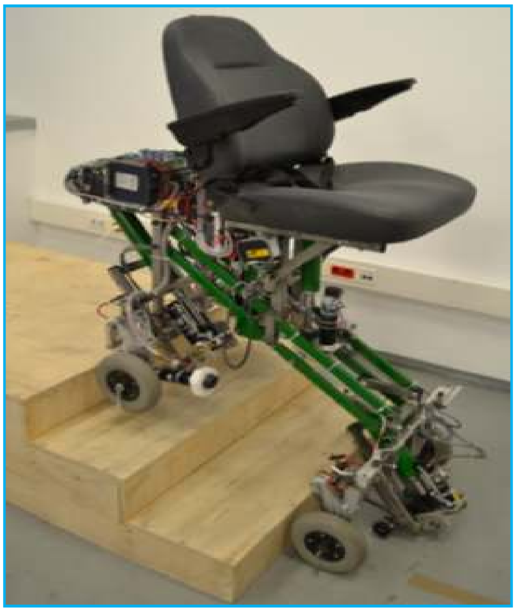


Figure 1.10: The Walking Wheelchair prototype.

- **Sharioto** (University of Leuven, Belgium, 2002, 2010)

It is equipped with infrared and ultrasonic sensors to measure surrounding obstacles, potentiometers for measuring the orientation of the front castor wheels, lidar, and a gyroscope for rotational velocity estimation. The main work focuses on the intention estimation (Figure 1.11) [22, 75].

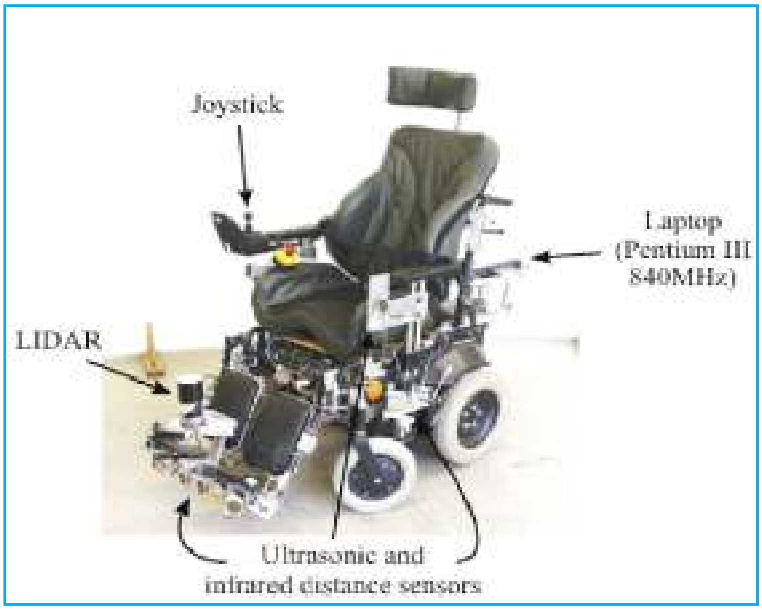


Figure 1.11: Sharioto prototype.

- **COALAS** (University of Picardie Jules Verne, France, 2012, 2015)

This project is focusing on designing and developing an autonomous cognitive platform combining an intelligent wheelchair (equipped with a set of heterogeneous sensors) and assistive capabilities of a humanoid robot. The project is structured in a user-centred co-creation process aiming to give responses to new needs and uses (Figure 1.12) [82].

Figure 1.12: COALAS prototype.

1.3. Sensors

To avoid obstacles, smart wheelchairs need sensors to perceive their surroundings. By far, the sensor most frequently used by smart wheelchairs is the sonar sensor (Figure 1.13a). Often, they are mounted in a ring around the wheelchair (e.g. Rolland), but sometimes they only cover the front of the vehicle (e.g. NavChair). Infra-red sensors are also often used (e.g. RobChair) (Figure I.13b). As they are relatively expensive, laser range finders are only sparsely used (e.g. Maid, SmartChair) (Figure 1.13c).

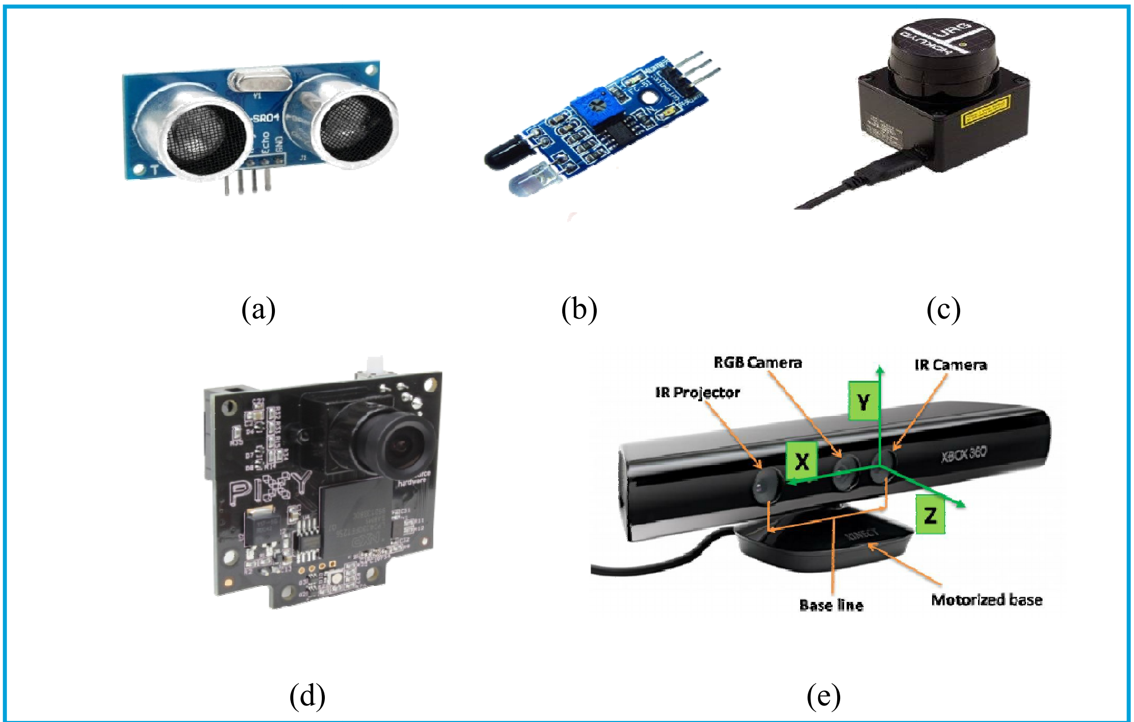


Figure 1.13: Examples of sensors used in IW applications: (a) sonar sensor, (b) infrared sensor, (c) laser range finders, (d) stereo camera, (e) inside Microsoft Kinect

Using a stereo camera and spherical vision system (Figure 1.13d) [55], 3D scanners like Microsoft's Kinect (Figure 1.13e), and most recently the Structure 3D scanner, it has become possible to use point cloud data to detect hazards like holes, stairs, or obstacles [19]. These sensors have until recently been relatively expensive, were bulky and would consume a lot of power.

1.4. Operating modes

There are three modes of operation: Manual, semi-autonomous, and autonomous. The manual mode simply uses the joystick to control the motors of the wheelchairs. Semi-autonomous and autonomous modes are detailed in the following section.

1.4.1. Semi-autonomous mode

The semi-autonomous mode allows users to move independently using integrated sensory information and human-machine interface. It's the usual way and also the best way to control a powered wheelchair if the user has the physical possibilities to do it.

1.4.2. Autonomous mode

Autonomous mode means that the user has an interface in which he selects a destination and the wheelchair calculates the path to get there. This method requires precise location of the chair in the environment but does not require much attention of the user. However, it asks whether the wheelchair has a complete knowledge of the environment using a map or that the environment has been modified by adding objects to it. In this case the operating mode does not allow navigating in an unknown environment [18].

1.5. The human machine interface

In the past decades, a lot of intelligent wheelchair control systems were developed using different methods and techniques. Human Computer Interface (HCI) and Human Machine Interface (HMI) are the latest and most effective techniques [15]. In user interface systems images, voices, motions, bio- signals, and tactile devices are used as a medium of control. In this section an overview of the most common methods used in human machine interface is given.

1.5.1. Vision Based Methods

Vision based methods can be used to recognize different types of human body motion. Some IW projects use these methods to capture the facial expressions, mouth gesture, gaze and position, hand gesture, and orientation of the patients' head to control the wheelchair.

1.5.1.1. Facial expression analysis

Facial expression analysis refers to computer systems that attempt to automatically analyze and recognize facial motions and facial feature changes from visual information. Some recent intelligent wheelchair projects presented input methods for people with quadriplegia, by using the recognition of facial expressions as the main input to guide the wheelchair [68].

1.5.1.2. Mouth gesture recognition

Mouth gesture recognition is another method which used to associate the users' intention to obtain desired outputs. A novel IW interface using face and mouth recognition for severely disabled individuals is proposed in [40]. For the recognition of the user's intention, the direction of the IW is determined according to the face inclination, while proceeding and stop are determined by the shape of the mouth.

1.5.1.3. Eye gaze tracking

In this technique, a camera is set up in front of wheelchair user to capture image information (Figure 1.14). The sequential captured image is interpreted to obtain the gaze direction and eye blinking properties image, which are used to provide the movement direction of IW and timing command, respectively [62].

Figure 1.14: Eye end gaze drive system.

1.5.1.4. Hand gesture recognition

Also hand gesture recognition is used as Vision based interface system for control of an IW [83]. In this system, Haar like features and adaBoost learning algorithm are used for hand gesture detection. The results of detection combined with centroid are used to determine control commands.

1.5.1.5. Head movement

Some vision based systems capture the orientation and position of the user's head, which can be used as an input to represent a specific desired output. This technique has been used in some wheelchair projects (eg. RoboChair) [38]. A combination of Adaboost face detection and camshift object tracking algorithm is used to achieve accurate face detection, tracking and gesture recognition in real time. By detecting frontal face and nose position, head gesture is estimated and used to control the wheelchair.

1.5.2. Voice Based Methods

Voice is also used as an input to control wheelchairs [4, 6, and 74]. In such systems voice recognition algorithms is used. When system recognizes the voice, it is classified to one of pre-stored commands and generates a signal to control the movements of wheels.

1.5.3. Motion Based Methods

Motion recognition is the interpretation of a human gesture by a computing device. In this context, motions are expressional movements of human body parts, such as: fingers, hands, arms, head, tongue, legs. The goal of motion recognition research is to develop systems which can identify specific human gestures and use them to send information for a device control.

1.5.3.1. Accelerometer

In accelerometer based method, the IW is controlled using head tilt. Firstly, the user tilts his head in front/back/left or right direction. Then, the accelerometer senses the change in direction of head and accordingly the signal is given to microcontroller. Depending on the direction of the acceleration, microcontroller controls the wheelchair [33].

1.5.3.2. Sip and Puff

In this method using air presser to generates control signals by sipping (inhaling) or puffing (exhaling) in a tube (Figure 1.15). This technology generates four control signals for motorized wheelchair which are initial hard puffs, hard sip, initial hard sip, and hard puff. It is mostly used for quadriplegics having injury in their spinal cord or people with ALS. But this is not good for individual with week breathing [79].

Figure 1.15: Sip and puff powered wheelchair

1.5.3.3. Tongue drive system

Tongue Drive System is a new idea that is developed by the team at Georgia Tech. This system uses two magnetic sensors placed on side of the users head and magnetic tongue barbell [42]. By placing the magnetic barbell inside specific sports of their mouth users are able to control the direction and speed of the wheelchair with better results than similar systems for users without the use of their hands (Figure 1.16).

Figure 1.16: Prototype Tongue drive system.

1.5.4. Bio – signals Based Methods

The bio-signals based methods like EMG, EOG or EEG are used for completely paralyzed patients who can only use their bio-signals as the only resource to control the IW.

1.5.4.1. Electromyography (EMG)

EMG measures electrical currents that are generated in muscles during its contraction. A muscle fiber contracts when it receives an action potential. The EMG observed is a result of the summation of all the action potentials that occur around the electrode site. In almost all cases, muscle contraction causes an increase in the overall amplitude of the EMG. EMG signals can be used for a variety of applications including clinical applications [15, 67]. They are easy to acquire and of relatively high magnitude than other bio-signals. On the other hand, EMG signals are easily susceptible to noise. They also have different signatures depending on age, muscle development, motor unit paths, skin fat layer, and gesture styles. The external appearances of two individuals' gestures might look identical, but the EMG signals characteristics are different [77].

1.5.4.2. Electrooculography (EOG)

EOG measures eye muscle movements, by placing electrodes above the eye (Figure 1.17). EOG signals records the potential difference between the retina and cornea of the eye. When the eyes are rolled upward or downward, positive or negative pulses are generated. As the rolling angle increases, amplitude of pulse also increases and the width of the pulse is indirectly proportional to the eyeball rolling process [17, 80].

Figure 1.17: Placement of electrodes for EOG measurement.

1.5.4.3. The Electroencephalography (EEG)

EEG is also one of the IW control methods as shown in (Figure 1.18). A set of electrodes is applied on the scalp and wired to an amplifying-filtering-digitalizing device to measure and to interpret the electrical activity of the brain. Then, it translates the brain activity in commands to control devices. EEG techniques are non-invasive and low cost. However, it brings great challenges to signal processing and pattern recognition, since it has relatively poor signal-to-noise ratio and limited topographical resolution and frequency range [16, 43].

Figure 1.18: EEG control IW

1.5.5. Smartphone and Tablet Based Methods

Smart phones and tablet computers are increasingly becoming popular. They are equipped with many useful built-in sensors. Researchers are recently using these devices for powerful controlling applications. Wheelchair controlling is one of these applications [8, 57, and 66]. Smartphone and its built-in sensors are used to capture and record physical activity of wheelchair users and translate it as orders to a controlling unit which control the chair. For example Authors in [66] use the 3 axis accelerometer present in most smart phones and Bluetooth wireless technology to allow patients to move the wheelchair by tilting the smart phone.

Authors in [57] designed a wheelchair monitored with android mobile to help the disabled patients by using an android application to control the movement of a wheelchair in different directions. The system presented in [8] used voice recognition to control the WI trough an android application.

1.6. Conclusion

In this chapter, after a brief introduction to the most popular IW projects and the necessary requirements for such project, we have presented an overview of some low cost input devices that can be used in IW systems, as well as the operating modes of IW projects. Then, the most popular mode of human machine interface is given at the end of the chapter, within this framework; some of these methods are described and used in this project, while the others were also briefly described to serve as examples.

The following chapters present the human machine interfaces which are used in our project.

CHAPTER 2

SPEECH-BASED HUMAN MACHINE INTERFACE SYSTEM

2.1. Introduction

Speech is the primary and the most important way of communication between human. By the developments of communication technologies in the last era, the speech starts to be an important interface for many systems. Instead of using complex different interfaces, speech is easier to communicate with computers.

In this chapter, an HMI system based on speech recognition is implemented using two different ways. The first way is using Microsoft Windows speech Recognition, which comes built into Windows 7, and Microsoft visual studio C#. The second way is using an android application in Smartphone which use Google Speech to Text to recognize and process human speech. In both methods, the chosen commands were tested by two users ten times in silent environment and ten times in noisy environment. The result shows that the overall accuracy is quite satisfactory.

This Chapter is organized as follow: In section 2.2, we briefly defined the speech recognition. Section 2.3 presents the classification of speech recognition system. Section 2.4 shows the principal Components of speech recognition system. Few speech recognition softwares are presented in Section 2.5. Section 2.6 describes our HMI system based on speech recognition and shows experimental results. Finally, Section 2.7 concludes the chapter.

2.2. Speech recognition

Speech recognition is the process of converting an acoustic signal (person's utterance), captured by micro- phone or a telephone, to a digitally stored set of words [41]. These words are later on recognized by speech recognizer, and in the end, system outputs the recognized words as written text.

The quality of a speech recognition systems are assessed according to two factors: its accuracy (error rate in converting spoken words to digital data) and speed (how well the software can keep up with a human speaker). Speech recognition technology has endless applications. Commonly, such voice-mail systems in telephony, hands-free machine operations, communication interfaces for people with special abilities, dictation systems, and translation devices.

2.3. Classification of speech recognition system

Speech recognition systems can be separated in several different classes [36].

According to the continuity of the speech:

- **Isolated words speech**
- **Keyword spotter**
- **Connected word**
- **Continuous speech**
- **Speech Understanding**

An isolated word recognition system is trained for the fixed set of words or phrases which comprise its vocabulary. These words must be spoken in isolation, i.e., with pause between successive inputs. In contrast, a keyword spotter is able to locate and recognize keywords (or phrases) which are embedded in conversational speech.

The distinction between connected-word recognizer and continuous speech recognition systems arises from the way in which the continuous utterances are modeled. A connected word recognizer uses words as recognition units, which can be trained in an isolated word mode. Continuous speech is generally connected to large vocabulary recognizers that use subword units such as phones as recognition units, and can be trained with continuous speech. If a continuous speech recognition system is even able to interpret the recognized utterances, then, unless the interpretation is trivial, it is termed a Speech Understanding system.

Besides being isolated, keyword, connected or continuous, speech recognitions systems are divided into three categories according to dependence on speaker:

- **Speaker Dependent**
- **Speaker Adaptive**
- **Speaker Independent**

Speaker independent recognition is more difficult, because the internal representation of the speech must somehow be global enough to cover all types of voices and all possible ways of pronouncing words, and yet specific enough to discriminate between the various words of the vocabulary.

For a speaker dependent system the training is usually carried out by the user, but for applications such as large vocabulary dictation systems this is too time consuming for an individual user. In such cases an intermediate technique known as speaker adaptation is used. Here, the system is bootstrapped with speaker-independent models, and then gradually adapts to the specific aspects of the user.

Another classification can be made according to the vocabulary size. Speech recognition systems can be separated into three groups:

- **Small vocabulary**
- **Medium vocabulary**
- **Large vocabulary**

Small vocabulary systems for command and control applications normally have an active vocabulary of up to a few tens words. Large vocabulary systems, on the other side, recognize a thousand words and more.

- Small vocabulary : tens of words
- Medium vocabulary : hundreds of words
- Large vocabulary : thousands of words

2.4. Component of a speech recognition system

A speech recognition system operates in two phases: training phase and recognition phase. In the training phase, both the acoustic model and the language model are trained

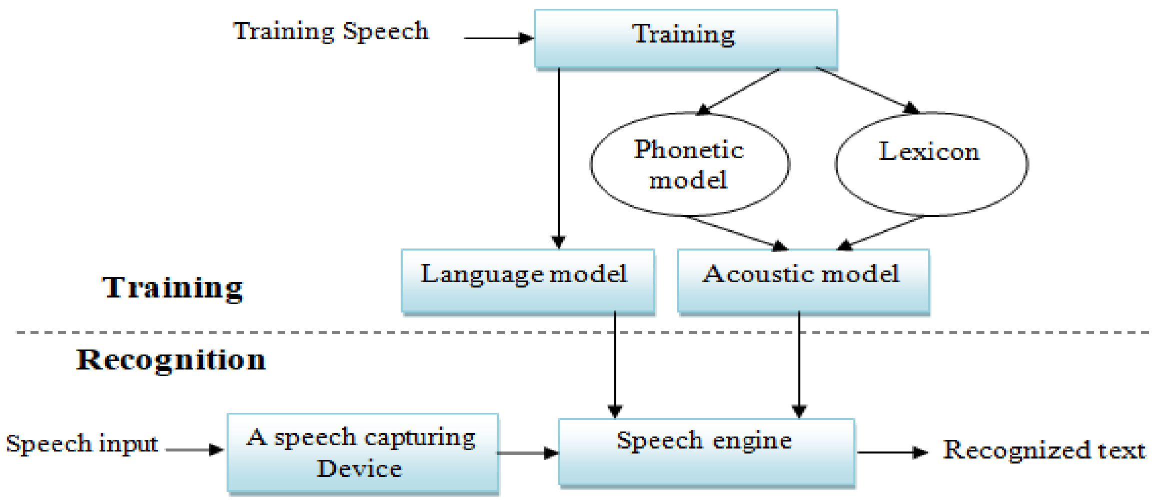


Figure 2.1: Speech recognition process.

2.4.1. Speech capturing Device

It consists of a microphone, which converts the sound wave signals to electrical signals and an Analog to Digital Converter which samples and digitizes the analog signals to obtain the discrete data that the computer can understand.

2.4.2. Training

Training is the process of estimating the speech model parameters from actual speech data. In preparation for training, what are needed are the text of the training speech and a lexicon of all the words in the training, along with their pronunciations, written down as phonetic spellings. Thus, a transcription of the training speech is made by listening to the speech and writing down the sequence of words. All the distinct words are then placed in a lexicon and someone has to provide a phonetic spelling of each word [49].

2.4.3. Acoustic model

Acoustic model contains a statistical representation of the distinct sounds that make up each word in the language Model or Grammar. Each distinct sound corresponds to a phoneme.

2.4.4. Language model or Grammar

Language Models contain a very large list of words and their probability of occurrence in a given sequence. They are used in dictation applications. Grammars are a much smaller file containing sets of predefined combinations of words. Grammars are used in command and control applications. Each word in a Language Model or Grammar has an associated list of phonemes (which correspond to the distinct sounds that make up a word).

2.4.5. Speech engine

Speech Recognition Engine uses software that is called Decoder, which get the sounds spoken by a user and finds the acoustic Model for the same sounds, when a match is completed, the Decoder determines the phoneme corresponding to the sound. It keeps track of the matching phonemes until it reaches a pause in the users' speech. It then searches the Language Model or Grammar file for the same series of phonemes. If a match is made it returns the text of the corresponding word or phrase to the calling program [73].

2.5. Few speech recognition softwares/applications

The following list presents notable speech recognition softwares that operate in Windows, Linux, Mobile devices and smartphones.

Application name	Operating system
Kaldi	Linux
Speech SDK	Windows 2000 and XP
Windows Speech Recognition	Come built into Windows vista,7,and 8
SpeechMagic	Windows7,8,and 10
Cortana	Come built into Windows 8.1, 10, and android
Google Now	Android

Table 2.1: List of speech recognition software [77].

2.6. Voice recognition based HMI system

HMI system based on speech recognition is implemented using two different ways. The first way is using Microsoft Windows Speech Recognition (MWSR), which come built into windows 7, and Microsoft visual studio C#. The second way is using an android application in Smartphone which uses Google Speech to Text to recognize and process human speech.

2.6.1. Speech control interface using MWSR

The process of implementing speech recognition to a Windows Forms application starts by launching MS Visual Studio and creating a new C# Windows Forms application with the proposed name. Before creating any program with speech recognition system in C#, "System.speech" library has to be added by "using" namespace statement at the beginning of the code. Then, two instances of the "SpeechRecognitionEngine" were created. The first instance is used to recognize the commands which control the interface (starting or stopping the speech recognition of the second instance, while the latter is used to recognize the commands which controls the direction of IW. The grammars also must be added into the "SpeechRecognitionEngine". Otherwise, the speech recognizer will not be able to recognize the targeted phrases. The command words included in both grammar lists are presented in Table 2.2

	Commands	Action
Operating interface data base	Démarrer le système	To start controlling the IW using speech recognition
	bad' alnizam (بدء النظام)	
	arrêter le système	To stop controlling the IW temporarily
	awqaf alnizam (أوقف النظام)	
IW control data base	droite	To turn right
	Yamine (يمين)	
	gauche	To turn left
	Yassare (يسار)	
	arrêt	To stop the movement
	Quf (قف)	
	avance	To move forward
	Amème (أمام)	
	arrière	To move backward
	Waraa (وراء)	

Table 2.2: The command words included in both grammar lists.

The execution of the program starts by reading the voice captured by microphone and converting it into words. These words firstly are compared only with operating interface data base. However, the recognizing of the command words which control the movement of IW will not commence unless the phrase "Démarrer le système" or "بدء النظام " is spoken by the user. Once one of these phrases is spoken, the system will be waiting for an utterance to analyze and compare with IW control data base. In the case of match with anty control word of the second grammar, the appropriate instruction will be sent to the microcontroller via the serial port. The flowchart of our system is shown in Figure 2.2

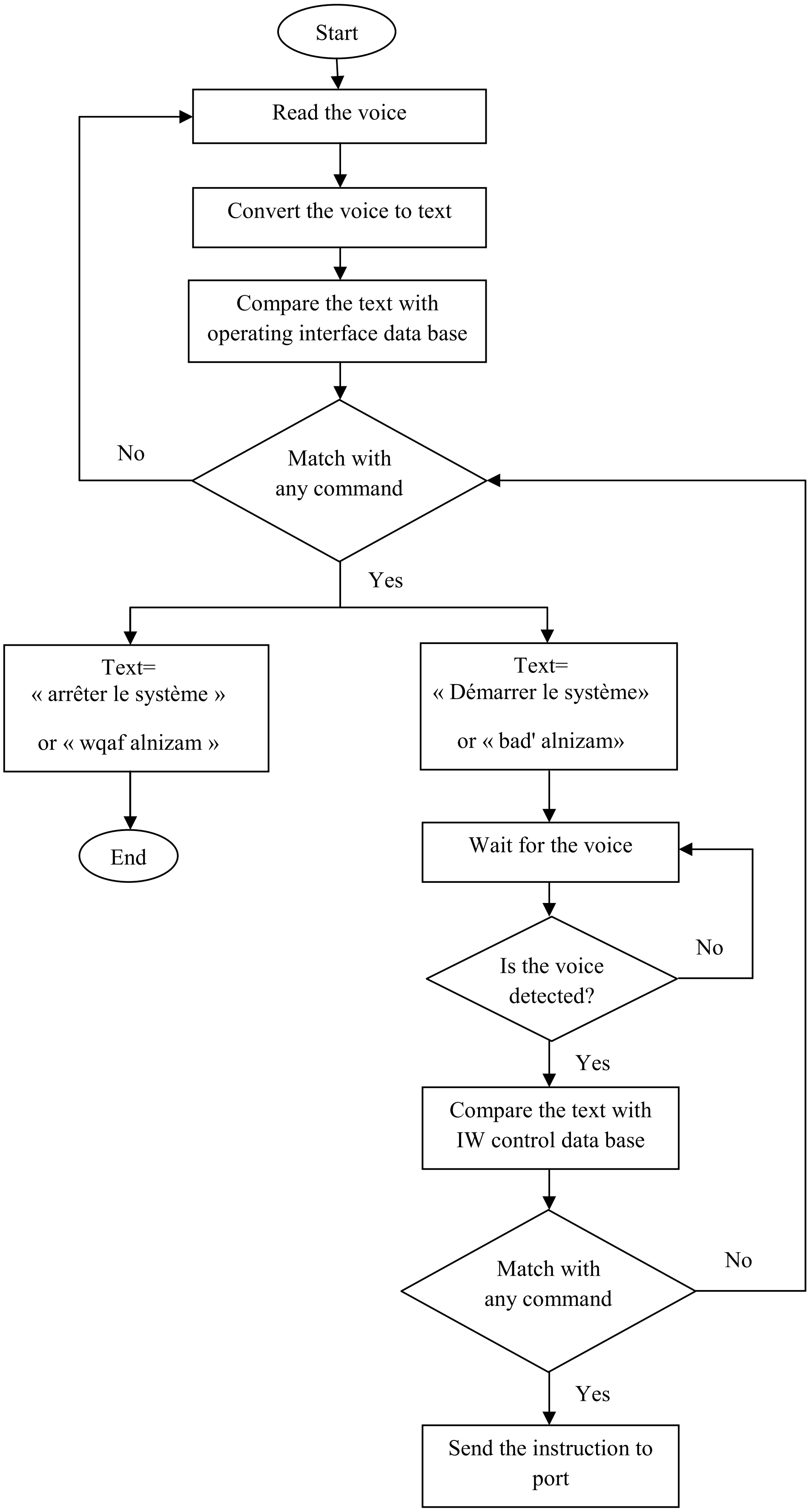


Figure 2.2: The flowchart of the system

2.6.1.1. User interface

The design of user interface provides all the information about the recognized commands and the control button of the system (Figure 2.3). The control over the user interface includes Start/Stop (commencer/ arrêter) button that tells the system to start or to stop temporarily recognizing the words which control the IW movement. The user can also do these steps by either vocal command, saying Démarrer le système" or " بدء النظام " to start the control of IW and " arrêter le système " or "أوقف النظام " to stop the control of IW, or by clicking on Start/Stop button.

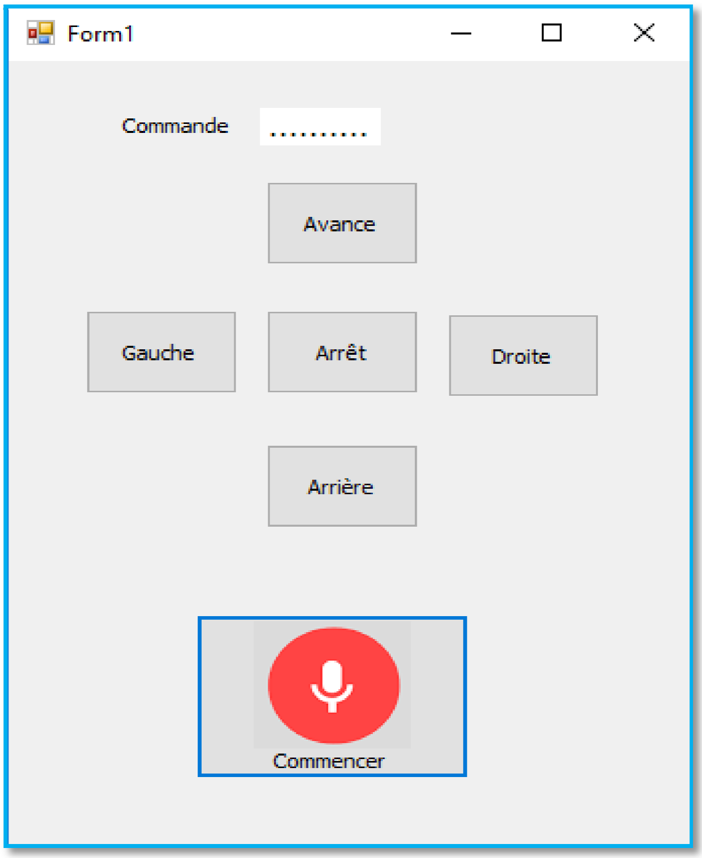


Figure 2.3: User interface.

2.6.1.2. Testing of system

The speech control system works in two phases: the training phase and the recognition phase. First, we trained our speech recognition engine through windows speech recognition because it's impossible to train it from code. Then, we went to recognition phase; the tested commands which presented in Table 2.2 were tested with the current user, who train the system, and other user. To ensure the accuracy of the system we asked each user to repeat each tested commands 10 times in silent environment and 10 times in noisy environment.

2.6.1.3. Results

Table 2.3 presents the results of the system for the both user in silent environment

Commands	Current user	Other user
Démarrer le système	10	10
arrêter le système	10	10
Droite	10	8
Gauche	10	10
Arrête	10	10
Avance	10	10
Arrière	10	10
بدء النظام	10	10
أوقف النظام	10	9
يمين	10	10
يسار	10	10
قف	9	9
أمام	10	10
وراء	10	10
Total	**139**	**136**

Table 2.3: The results of the system for both users in silent environment

From Table 2.3, there are 139 over 140 commands recognized by our system for the current user and 136 over 150 commands recognized for the other user. It's clear that the success of the system for the other user is less than current user.

The percentage of the accuracy of the system in silent environment is 98.21%, calculation for percentage is shown as below.

$$accuracy = \frac{139 + 136}{280} \times 100\%$$

$$= 98.21 \ \%$$

Table 2.4 presents the results of the system for the both users in noisy environment

Commands	Current user	Other user
Démarrer le système	9	9
arrêter le système	9	9
Droite	9	7
Gauche	9	9
Arrête	8	8
Avance	9	8
Arrière	9	9
بدء النظام	9	8
أوقف النظام	8	8
يمين	9	9
يسار	9	9
قف	8	7
أمام	9	8
وراء	8	8
Total	**122**	**116**

Table 2.4 : The results of the system for both users in noisy environment

From Table 2.4, there are 238 over 280 commands recognized by the system for the both users.

The percentage of the accuracy of the system in silent environment is 85%, calculation for percentage is shown as below.

$$accuracy = \frac{122 + 116}{280} \times 100\%$$

$$= 85\%$$

Image 2.4 present the commands which control user interface and IW (“démarrer le système/ بدء النظام”, “droit”, “gauche”, “arrête”, “avance”, “arrière”,” يمين”,” يسار”,” قف ”,” أمام”,” وراء ”, and “arrêter le système /أوقف النظام ”.

.

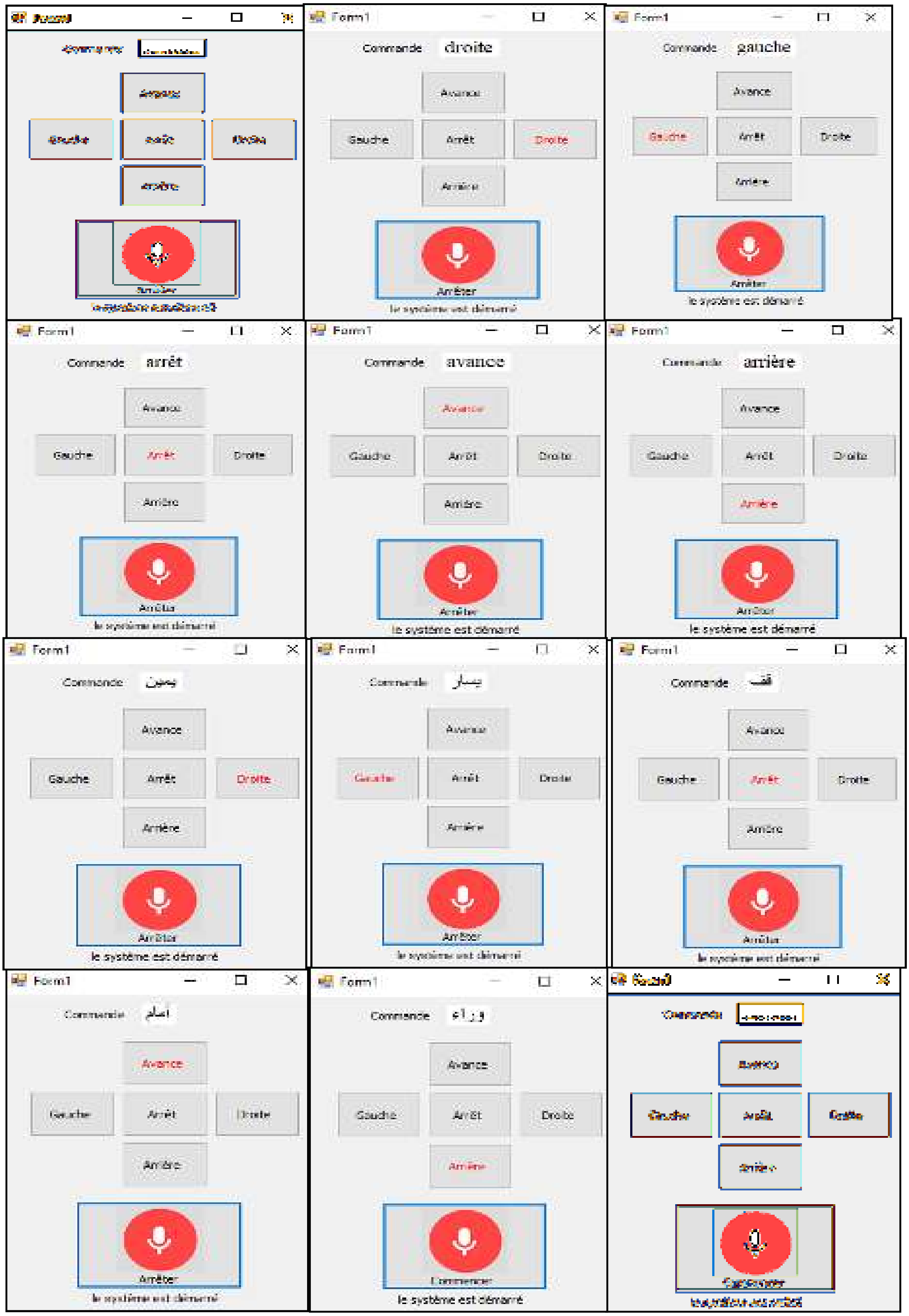


Figure 2.4: The results of commands in user interface.

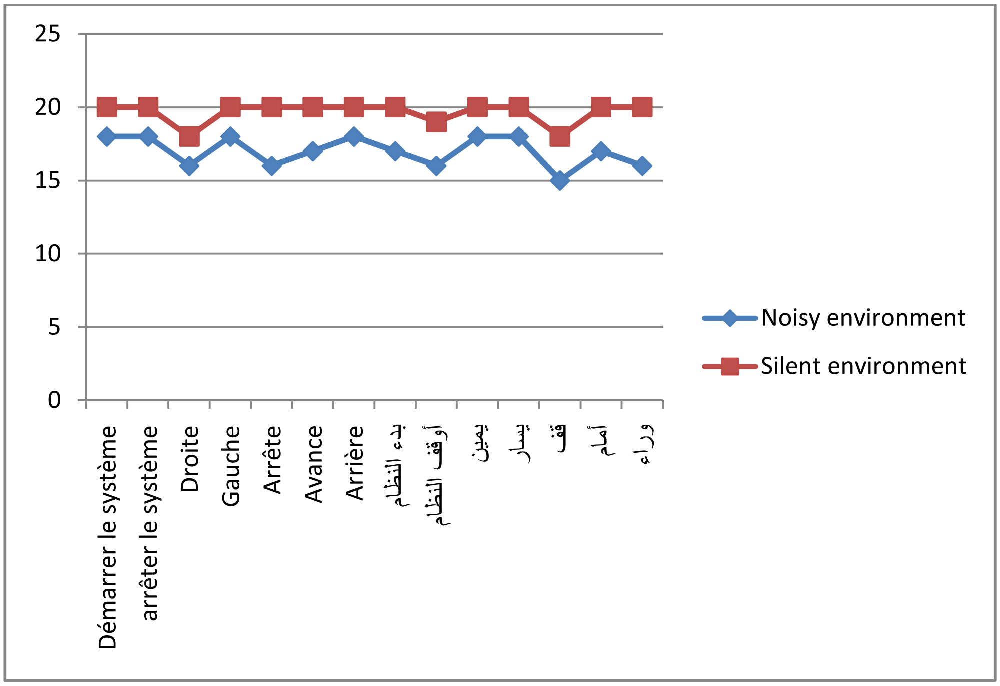


Image 2.5: the result of tested command in both environments

From the graph, we can find out that the system accuracy is less when assigning the commands in the noisy environment. That means the HMI based speech recognition has less control when in the noisy condition.

2.6.2. Speech control interface using android application

In this interface, the user could operate the wheelchair wirelessly using an android application in Smartphone which uses Google Speech to Text to recognize and process human speech.

The Android application is generally developed using JAVA language. The proposed application which will be used to control the wheelchair can be built without knowledge in java language. It is called as “Voice_controlled_wheelchair” developed using MIT App Inventor. Figure 2.6 shows below is a diagram which shows the interface of the application.

To be active this application requires an internet connection and Bluetooth connection to send the appropriate commands wirelessly.

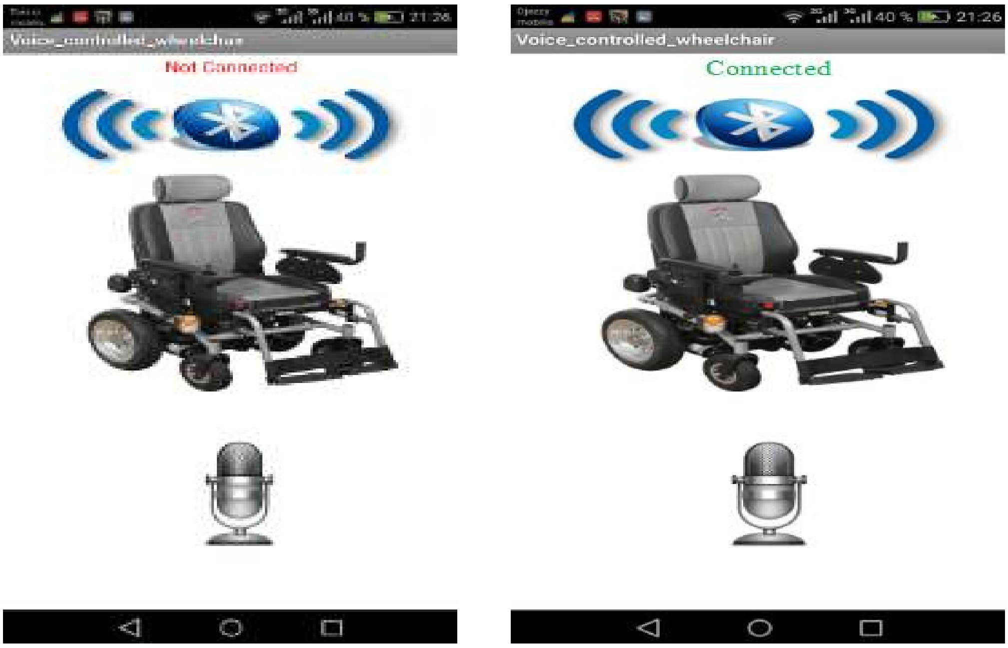


Image 2.6: Screenshots of the Voice_controlled_wheelchair

2.6.2.1. Description

Firstly, the user has to give vocal commands via android smart phone like ”avance”, “arrière”, “gauche”, “droite”, and “arrête“ These vocal commands are recognized and wirelessly transferred through the application via Bluetooth module as a data string to the microcontroller e.g. if the user says ”avance” the android phone sends the data string “* avance#” to your Bluetooth module *and # indicate the start and stop characters and can be used with any microcontroller which can handle strings. Then, the microcontroller processes this data strings and generate the appropriate control signal to send to the wheelchair control board.

2.6.2.2. Results

In order to test the accuracy and effectiveness of this interface two persons (male and female) were asked to repeat the vocal commands (avance, arrière, gauche, droite, and arrête) 10 times in silent environment and 10 times in noisy environment.

Table 2.5 presents the results of the system for the both user in silent environment

Commands	User 1	User 2
Avance	10	10
Arrière	9	9
Gauche	9	10
Droite	10	9
Arrête	9	9
Total	**47**	**47**

Table 2.5: The results of the system for both user in silent environment

From table 2.5, there are 94 over 100 commands recognized by the system for the both users.

The percentage of the accuracy of the system in silent environment is 94%, calculation for percentage is shown as below.

$$accuracy = \frac{47 + 47}{100} \times 100\%$$

$$= 94\%$$

Table 2.6 presents the results of the system for the both user in noisy environment

Commands	User 1	User 2
Avance	8	7
Arrière	8	8
Gauche	9	9
Droite	9	8
Arrête	8	9
Total	**42**	**41**

Table 2.6 : The results of the system for both user in noisy environment

From Table 2.6, there are 83 over 100 commands recognized by the system for the both users.

The percentage of the accuracy of the system in silent environment is 93%, calculation for percentage is shown as below.

$$accuracy = \frac{47 + 46}{100} \times 100\%$$

$$= 83\%$$

Image 2.7 presents the commands « avance », « arrière », « gauche », « droite », and « arrête ».

Image 2.7: The results of commands in android application.

2.7. Conclusion

In this Chapter, we implemented an HMI system based on speech recognition with two different ways. The first way is using Microsoft Windows speech Recognition and Microsoft visual studio C#. The second way is using an android application in Smartphone which uses Google Speech to Text to recognize and process human speech. In both methods, the system is tested many times by two persons (the current user and another user) once in a silent environment and once in a noisy environment. Experimental results show that the accuracy of the system in a silent environment is better than it in the noisy environment.

CHAPTER 3

VISION-BASED HUMAN MACHINE INTERFACE SYSTEM

3.1. Introduction

Vision based method can be used to recognize different types of human body. Some IW projects have used these methods to capture the facial expressions, mouth gesture, gaze and position, hand gesture, the orientation of the patient's head to control the wheelchair.

In this chapter, two Vision based interface systems for hands free control of an intelligent wheelchair are presented. The first interface uses the head gesture to control the movement of the wheelchair. Firstly, the head is detected using haar cascade. Once, the initial tracking window is determined; the new head location is determined using Camshift algorithm. Then, the head gesture command is determined according to the position of a rectangle containing the patient head in the image. The second interface uses mouth gesture. This system consists of two main parts: mouth detection using template matching and command extraction which is determined according to the detected gesture (Open mouth, Tongue on the left, tongue on the right).

The remainder of this chapter is organized as follows: Section 2 and section 3 describe our HMI system based on hand gesture and mouth gesture respectively and show experimental results. A brief conclusion is proposed in section 4.

3.2. HMI system based on head gesture recognition

As shown in Figure 3.1, the proposed system consists of two main parts. The first part is the face tracking. After image acquisition from camera, an initial face area is detected using haar cascade .Once, the initial tracking window is determined; the new head location is determined using Camshift algorithm. The second part is the command extraction. The controlling command is determined according to head gesture determination.

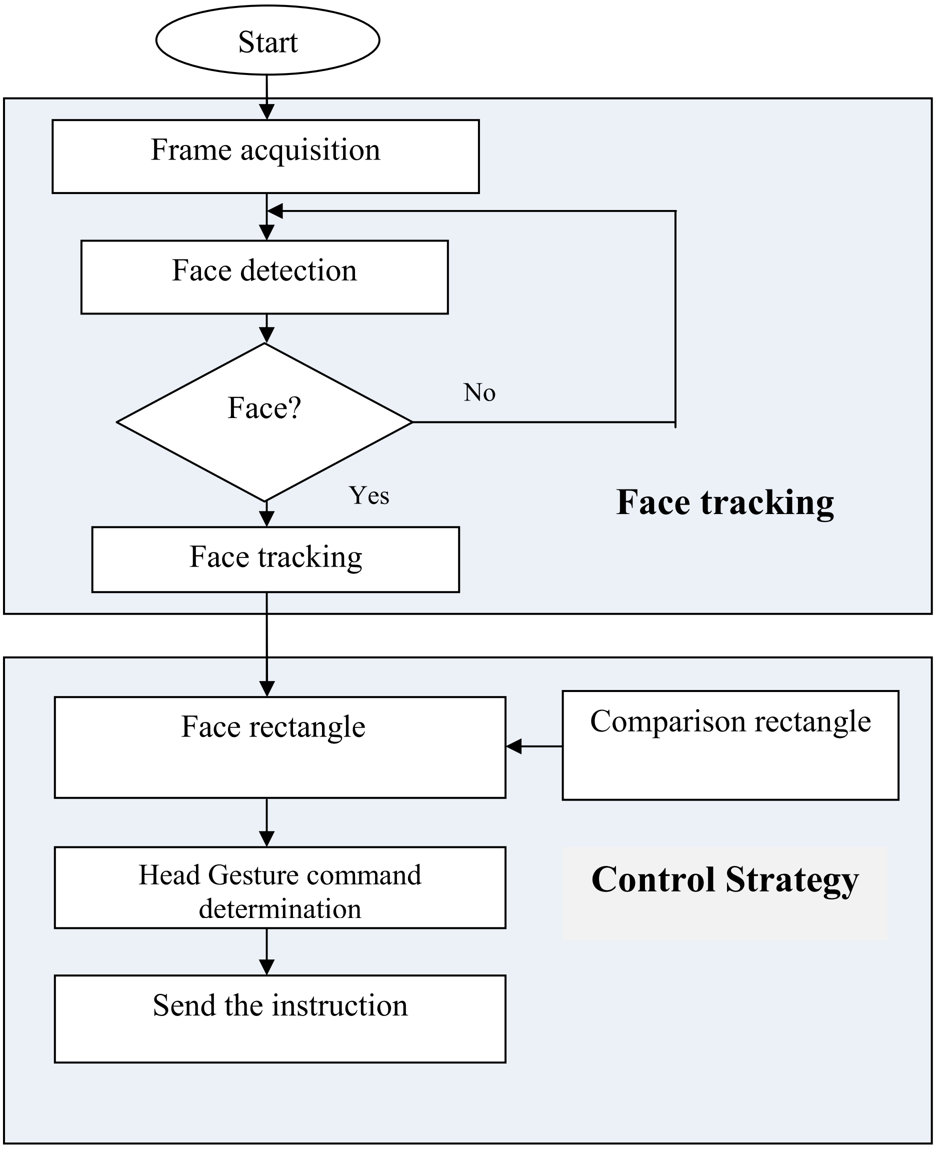


Figure 3.1: HMI system based on head gesture recognition structure.

3.2.1. Head detection

The main function of this step is to determine whether user face appears in a given image, and where this face is located at. For our project, Open Source Computer Vision Library (OpenCV) [7, 11, and 12] is used to implement the haar cascade classifier. Object detection using haar cascade classifier is an effective object detection method proposed by Paul Viola and Michael Jones [76]. It is a machine learning based approach where a cascade function is trained from a lot of positive images (which contain the object that want to detect it) and negative images (which do not contain the object that want to detect it). Then, it is used to detect objects in other images.

Haar features are the main part of the haar cascade classifier. They are used to detect the presence of feature in given image. Each features result in a single value which is calculated by subtracting the sum of pixels intensities in white region from that in black region as shown in equation (3.1) . Figure 3.2 shows some examples of haar like features.

$$P(x) = \sum Black\ pixel\ value - \sum White\ pixel\ value \qquad (3.1)$$

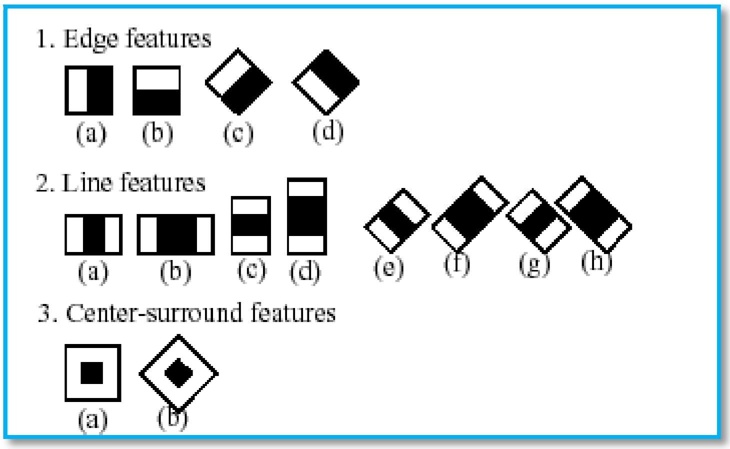


Figure 3.2: Haar features

The Haar feature starts scanning the image for the detection of the face from the top left corner and ends the face detection process bottom right corner of the image as shown in Figure 3.3. The image is scanned several times through the haar like features in order to detect the face from an image.

Figure 3.3: Image is scanned from top left corner to the bottom right corner.

To compute the rectangle features rapidly integral image concept is used [27]. It needs only four values at the corners of the rectangle for the calculation of sum of all pixels inside any given rectangle. In an integral image the value at pixel (x, y) is the sum of pixels above and to the left of (x, y). Sum of all pixels value in rectangle D is shown in Figure 3.4:

$$\begin{cases} sum(A) = val(1) \\ sum(A+B) = val(2) \\ sum(A+C) = val(3) \\ sum(A+B+C+D) = val(4) \\ \\ sum(D) = val(4) - val(3) - val(2) + val(1) \end{cases}$$

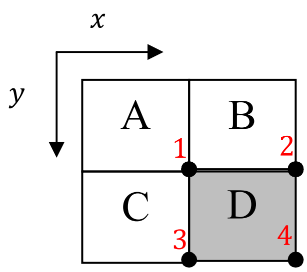


Figure 3.4: Calculation of integral image

Voila Jones algorithm uses a 24×24 window as the base window size to start evaluating these features in any given image. If we consider all the possible parameters of the haar features like position, type and scale, then we have to calculate about 160, 000 features in this window but this is practically impossible. The solution of this problem is to use the adaboost algorithm. Adaboost is a machine learning algorithm which helps us to find only the best features among the 160,000. These features are the weak classifiers. Adaboost construct a strong classifier as a linear combination of weighted simple weak classifiers as shown in equation (3.2).

$$F(x) = \alpha_1 f_1(x) + \alpha_2 f_2(x) + \cdots\cdots\cdots \qquad (3.2)$$

Where

x : Image

F : Strong classifier

f_n : Weak classifiers

α_n : Weight of weak classifiers

The face detection can be accomplished by cascade using haar like features as shown in Figure 3.5. In that cascade, an image will be a face if it passes all the stages. If it fail to pass any one of th stages it means that the image is not a face.

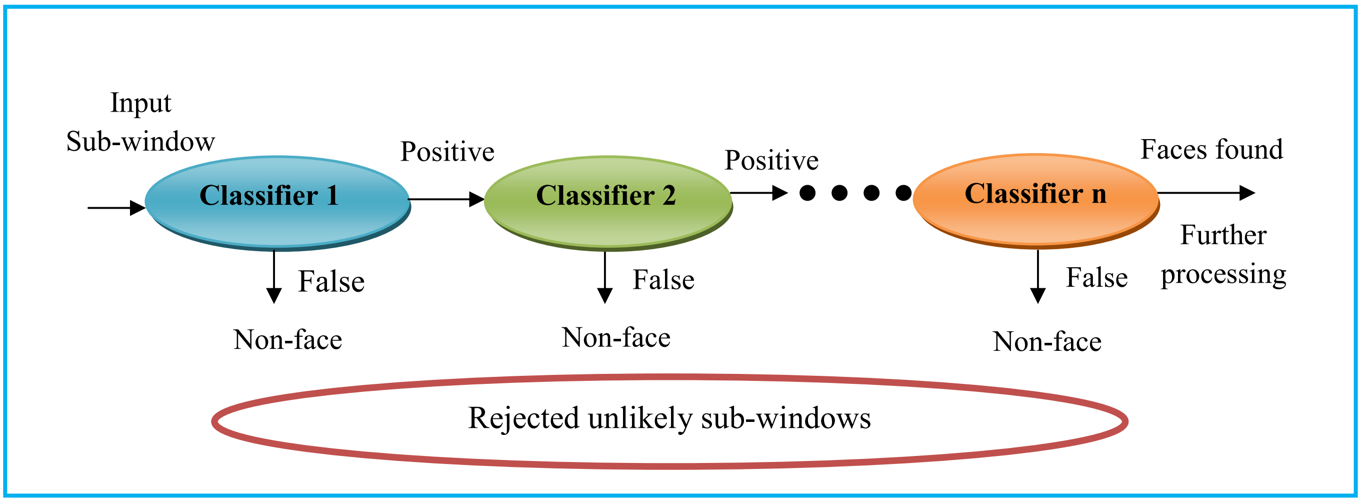


Figure 3.5: Cascade classifier

3.2.2. Head tracking

The goal of head tracking system is simply to know where the head constantly is in the image. Head tracking is composed of two steps: firstly tracked head is selected using the first location of the face which is detected by haar cascade classifier and afterwards Camshift algorithm is processed to track the head in the image [10, 25].

The CAMShift algorithm is an adaption of the Mean Shift algorithm for object tracking [18, 20]. The Mean Shift algorithm is a robust, non-parametric technique that climbs the gradient of a probability distribution to find the mode (peak) of the distribution. The primary improvement of CAMShift is that CAMShift uses continuously adaptive probability distributions (i.e., distributions that may be recomputed for each frame) while Mean Shift is based on static distributions, which are not updated unless the object being tracked significant changes in shape, size or color. The CAMShift procedure can be summarized in five steps.

1: Set the region of interest (ROI) of the probability distribution image to the entire image.

2: Select an initial location of the Mean Shift search window. The selected location is the target distribution to be tracked.

3: Calculate the color probability distribution of the region centered at the Mean Shift search window.

4: Iterate Mean Shift algorithm to find the centroid of the probability image. Store the 0th moment (distribution area) and centroid location.

5: For the following frame, center the search window at the mean location found in step 4 and set the window size to a function of the 0th moment. Then go to Step 3.

The creation of the probability distribution function corresponds to steps 1 to 3. For initialization, we use the face region detected on the last frame in detection step as the initial location of the search window. Then, we need to calculate the color histogram corresponding to this region in HSV color space. Since human's skin colors have little difference in S and V channels, only the H (hue) channel is used to compute the color distribution, which consumes the lowest number of CPU cycles possible. The histogram is quantized into bins to reduce the computational and space complexity and allow similar color values to be clustered together. Then a histogram back-projection is applied. Histogram Back-Projection: It is a primitive operation that associates the pixel values in the image with the value of the corresponding histogram bin. In all cases the histogram bin values are scaled to be within the discrete pixel range of the 2D probability distribution image using equation (3.3).

$$\left\{\hat{p}_u = \min\left(\frac{255}{\max(\hat{q})}\hat{q}_u, 255\right)\right\}_{u=1,\ldots m} \quad (3.3)$$

Where $\hat{q}_u$ is the unweighted histogram corresponding to the $n-th$ bin, and m is the numbers of bins. That is, the histogram bin values are rescaled from [0; $max(q)$] to the range[0; 255], where pixels with the highest probability of being in the sample histogram will map as visible intensities in the 2D histogram backprojection image.

Mass Center Calculation: The mean location (centroid) within the search window of the discrete probability image computed in Step 3 is found using moments. Given the intensity of the discrete probability image I(x; y) at (x; y) within the search window, the mass center is computed from:

$$M_{00} = \sum_x \sum_y I(x,y) \quad (3.4)$$

$$M_{10} = \sum_x \sum_y xI(x,y) \quad (3.5)$$

$$M_{01} = \sum_x \sum_y yI(x,y) \quad (3.6)$$

$$x_c = \frac{M_{10}}{M_{00}};\ y_c = \frac{M_{01}}{M_{00}} \quad (3.7)$$

Where M_{00} is the $zero^{th}$, M_{10} and M_{01} are the first moments for x and y respectively, $(x_c; y_c)$ is the next center position of the tracking window.

The Mean Shift component of the algorithm is implemented by continually recomputing new values of $(x_c; y_c)$ for the window position computed in the previousframe until there is no significant shift in position, i.e., convergence. Figure 3.6 shows one iteration of the Mean Shift algorithm.

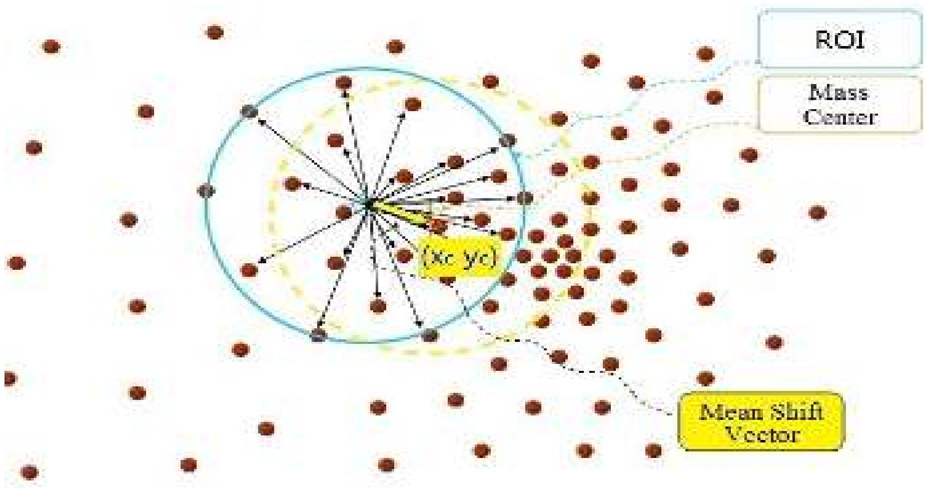


Figure 3.6: An Iteration of the Mean Shift Algorithm

3.2.3. Head gesture determination

The head gesture command is determined according to the position of the rectangle containing the patient head in the image, where the image plan is divided into five zones as shown in Figure 3.7. When the rectangle lies in the left side, the wheelchair will turn left, the neutral area means move backward, right side means turn right, top area means go forward and bottom area means stop moving.

<table>
<tr><td colspan="3">Up
(Forward)</td></tr>
<tr><td>Left
(Turn Left)</td><td>Center
(Neutral)</td><td>Right
(Turn Right)</td></tr>
<tr><td colspan="3">Down
(Backward) or (Stop)</td></tr>
</table>

Figure 3.7: Image plan partition

3.3. HMI system based on mouth gesture recognition

In this section, a novel IW interface using simple mouth gesture recognition is presented. As shown in Figure 3.8, the proposed system consists of two main parts. The first part is the mouth detection. After image acquisition from camera, a face area is detected using haar cascade classifier. Once the face is detected, Template matching is performed for detecting the mouth from lower face [48, 58]. The second part is the command extraction. The controlling command is determined according to mouth gesture determination.

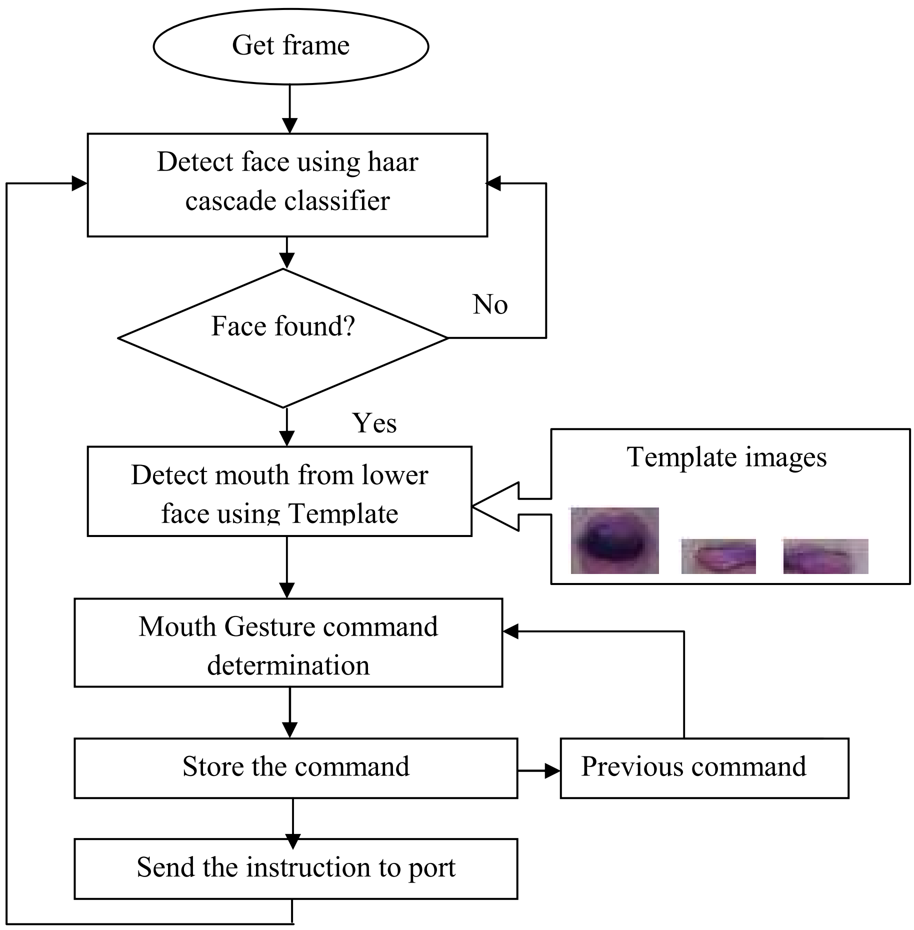


Figure 3.8: HMI system based on mouth gesture recognition structure.

3.3.1. Face detection

The main function of this step is to determine whether user face appears in given image, and where this face is located at to extract the approximate mouth region. For our project, Open Source Computer Vision Library (OpenCV) is used to implement the haar cascade classifier.

3.3.2. Mouth detection

After the system determines the approximate mouth region, template matching is performed for detecting the mouth. Template matching is a technique for finding areas of an image that match to a template image (Small part of image) [13]. Before performing template matching, we load three template images to the system (Figure 3.8). These images present the gestures that can be used by the mouth for controlling the wheelchair movement.

To apply the template matching technique, *MatchTemplate* an OpenCV function is used. This function requires three parametres, the first is the imput source image I, the image that contains what we are searching for. Here we selected the lower face image as input source image. The second is template image T; this image is contained within the input source image and we are looking to select its location. In our system we used three template images which mean that we applied *MatchTemplate* function three times, one for each template image in the frame. Finally, the third parametre is our template matching method, there are variety of methods available in OpenCV, but in our system we used the correlation coefficient which is specified by the flag CV_TM_CCOEFF_NORMED. The equation for this method is shown below:

$$R(x,y) = \frac{\sum_{x',y'}[T'(x',y').I'(x+x',y+y')]}{\sqrt{\sum_{x',y'}[T'(x',y')^2.I'(x+x',y+y')^2]}} \quad (3.8)$$

Where: $R(x,y)$ is numerical index for correlation coeficient

$T'(x',y')$ is the average value of template T

$I'(x+x',y+y')$ is the average value of I in the regin coincide with T

$x' = 0 \ldots\ldots\ldots. width - 1, y' = 0 \ldots\ldots.. high - 1$

Once the match was found in the source image, we excluded a rectange around the area corresponding to the highest match.

3.3.3. Mouth gesture determination

The mouth gesture command is determined according to which gesture from the three template images is detected (Open mouth, Tongue on the left,tongue on the right). In each frame, the used instruction which controlled the wheelchair movement is stored as provious command. Five commands are used to control the wheelchair and two other commands are used to stoping and starting the mouth gesture recognition controller, that mean seven gestures must be recognized by the system (Figure 3.9).

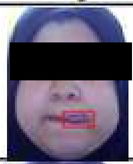
Mouth gesture	Time (s)	provious command	Control commands
	/	Left, right, stop, forward or backward	Turn Right
	/	Left, right, stop, forward or backward	Turn Left
	$t \leq 2\,s$	Left, right, forward or backward	Stop
		Stop	Go forward
	$2\,s < t \leq 7$	Left, right, stop, forward or backward	Go backward
	$t > 7\,s$	Left, right, stop, forward or backward	Stop controlling
		Stop controlling	Start controlling

Figure 3.9: Mouth gesture used to control the wheelchair

If the user take out his/her tongue on the left, the gesture is recognized as the left turn command.

If the user take out his/her tongue on the left, the gesture is recognized as the right turn command.

If the user open his mouth, the system triggers a timer to calculate the time taken befor closing his/her mouth (t)

- If $t \leq 2\,s$ and provious command is "Left, Right, Forward or Backward ", the gesture is recognized as Stop command.
- If $t \leq 2\,s$ and provious command is ''Stop'', the gesture is regonized as forward command.
- If $2\,s < t \leq 7\,s$ and provious command is "Stop", the gesture is regonized as backward command.
- If $t > 7\,s$ and provious command is '''Left, Right, Stop, Forward or Backward '', mouth gesture recognition controller will be Stopped.

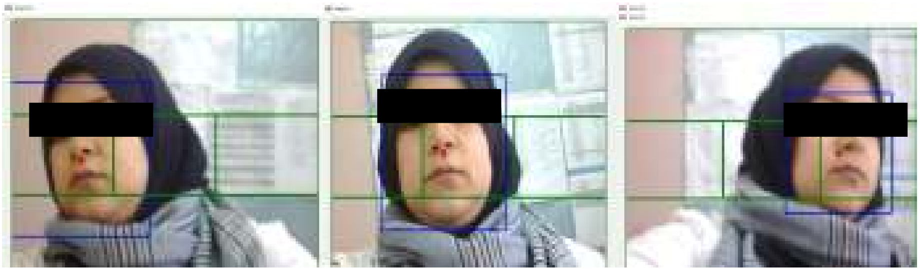
- If $t > 7\ s$ and provious command is ''Stop controlling'', mouth gesture recognition controller will be started.

3.4. Results

In order to test the accuracy and the effectiveness of the both interfaces, two persons (male and female) were asked to apply the commands twenty times in scenes which had cluttered stationary background with a varying illumination.

3.4.1. HMI system based on head gesture recognition

Figure 3.10 shows some head tracking results for both users, Figure 3.11 shows the head gesture recognition results in a bright light condition, Figure 3.12 shows the head gesture recognition results in a normal light condition, Figure 3.13 shows the head gesture recognition results in a dark light condition,



(a)

(b)

Figure 3.10: Head tracking results for both users: (a) subject 1,(b) subject 2.

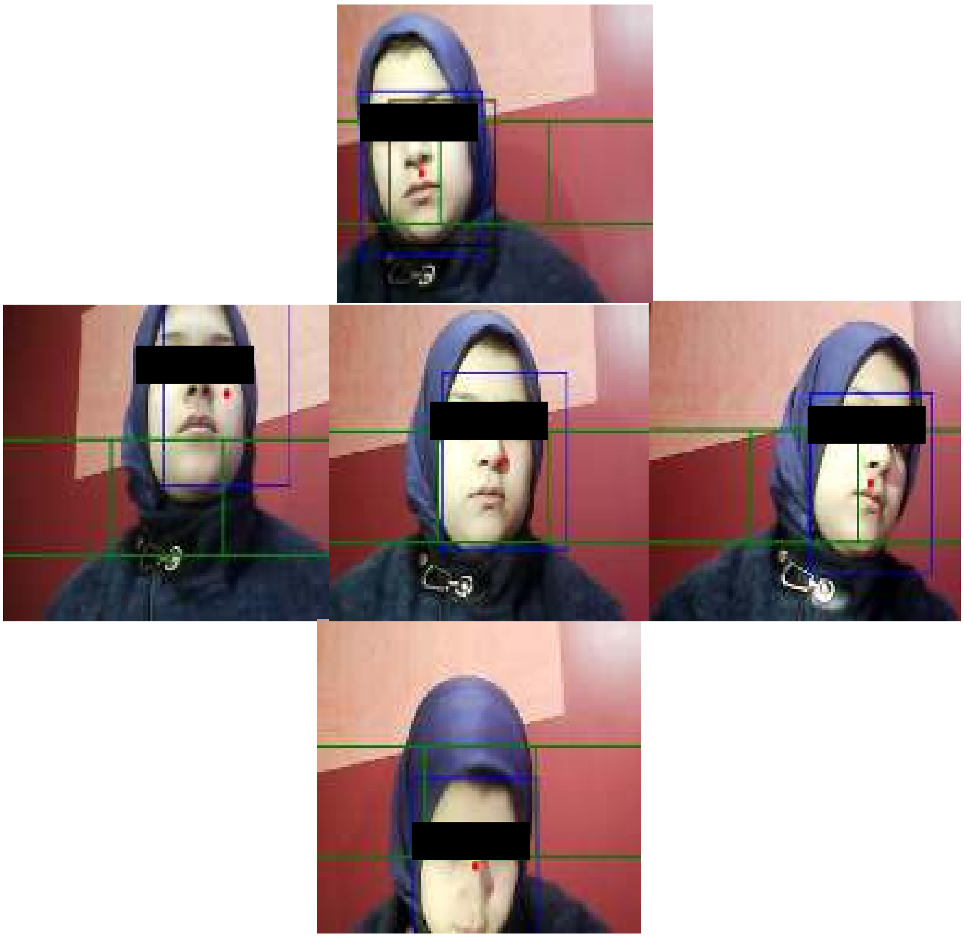


Figure 3.11 : Head gesture recognition results in a bright light condition.

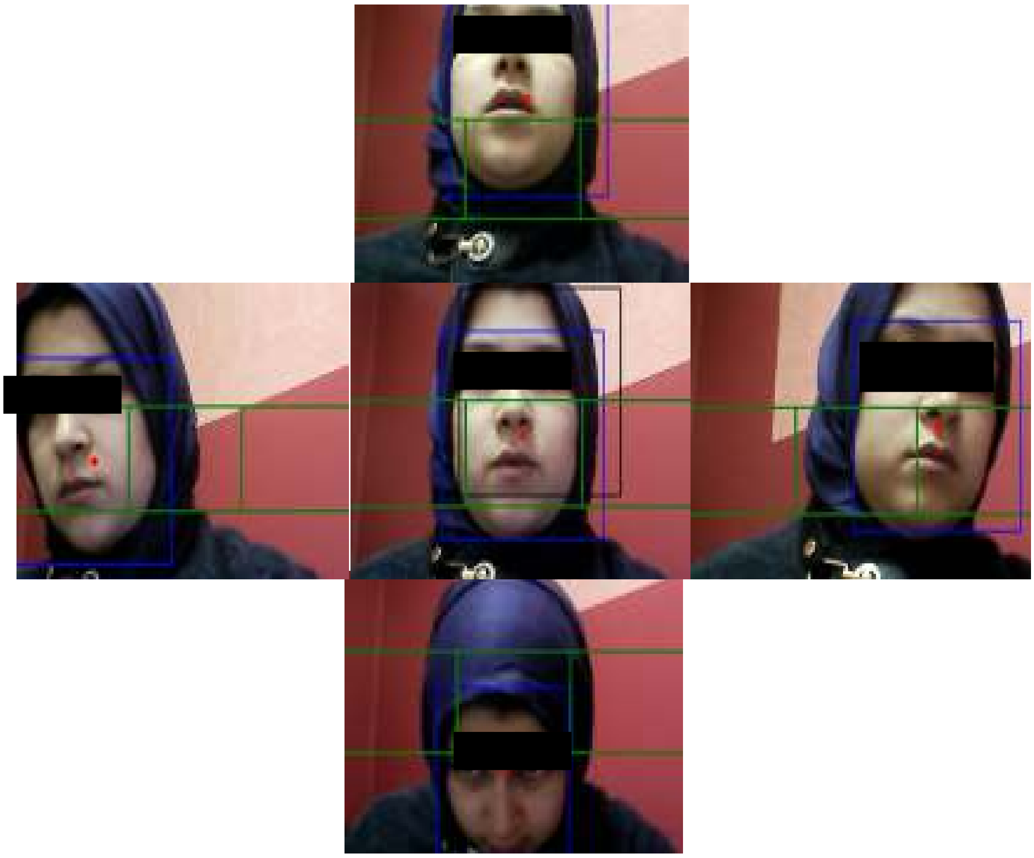


Figure 3.12: Head gesture recognition results in a normal light condition.

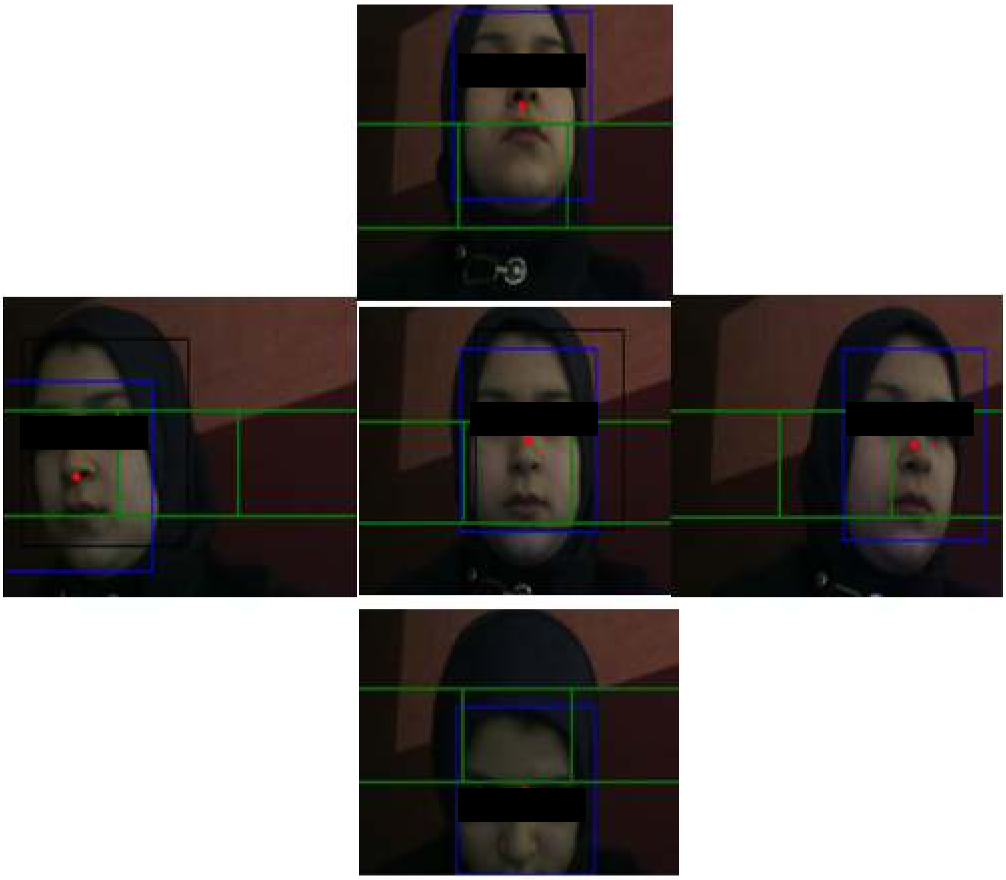


Figure 3.13: Head gesture recognition results in a dark light condition.

As indicated in the figures above, the detected face is drawn in black color, the tracking head is drawn in blue color, and the center of the rectangle containing the head area is drawn in red color. The five commands are tested in different lighting conditions and in complex background, which proves the invariance of the system to lighting condition and cluttered backgrounds.

Table 3.1 presents the results of the system for the both user in a (bright, normal, and dark light condition).

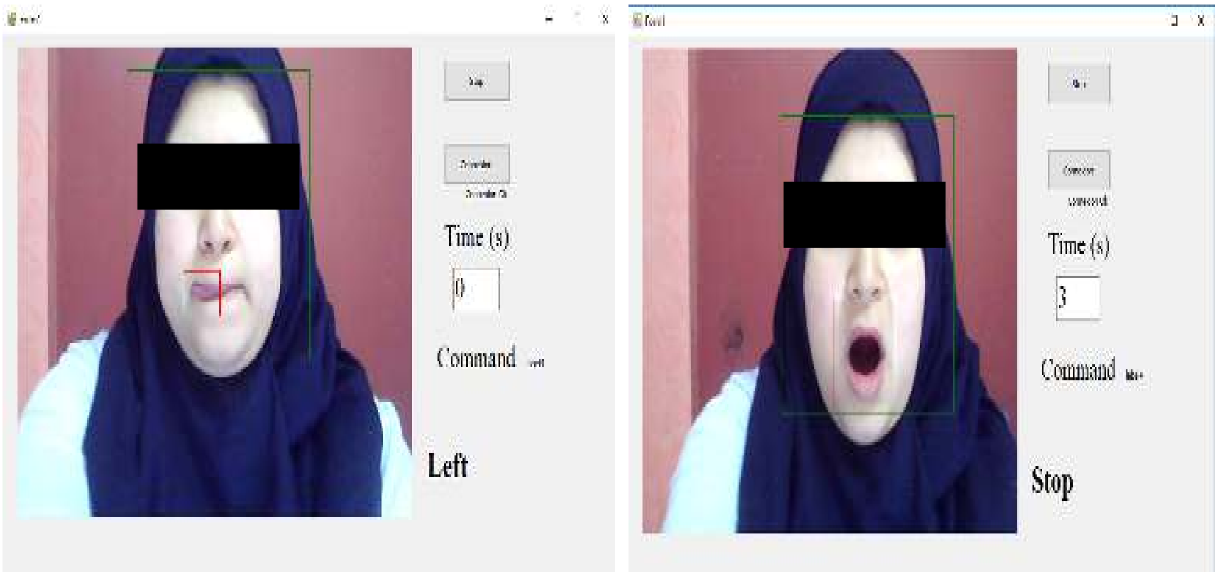
	Bright light condition		**Normal light condition**		**Dark light condition**	
Command	**User 1**	**User 2**	**User 1**	**User 2**	**User 1**	**User 2**
Forward	19	18	20	19	18	18
Backward	18	17	18	17	17	16
Left	20	19	20	20	17	18
Right	19	19	20	19	18	18
Stop	18	18	20	19	17	17
Total	**185**		**192**		**180**	
Accuracy	**92.5%**		**96%**		**87%**	

Table 3.1: The results of the system for the both user in different lighting conditions.

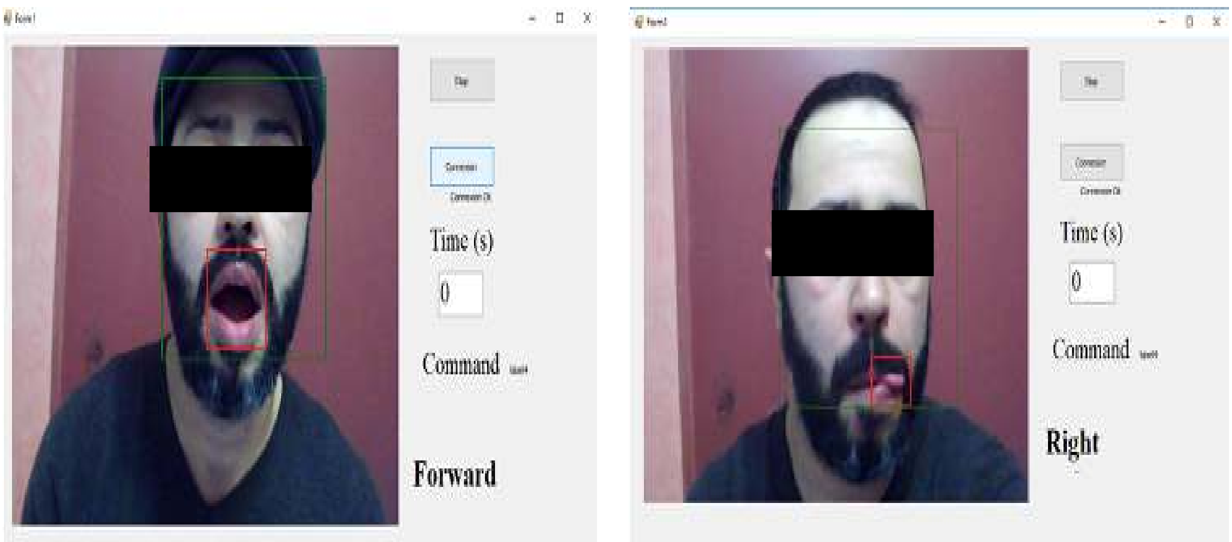
From the results, we can find out that the system accuracy is quite satisfactory. However, we found that the system performance is lower in both bright and dark lighting environment than in a normal light environment.

3.4.2. HMI system based on mouth gesture recognition

Figure 3.14 shows some mouth gesture recognition results for both users, Figure 3.15 shows the mouth gesture recognition results in a bright light condition, Figure 3.16 shows the mouth gesture recognition results in a normal light condition, Figure 3.17 shows the mouth gesture recognition results in a dark light condition.



(a)



(b)

Figure 3.14: Mouth gesture recognition results for both users: (a) subject 1,(b) subject 2.

Figure 3.15: Mouth gesture recognition results in a bright light condition.

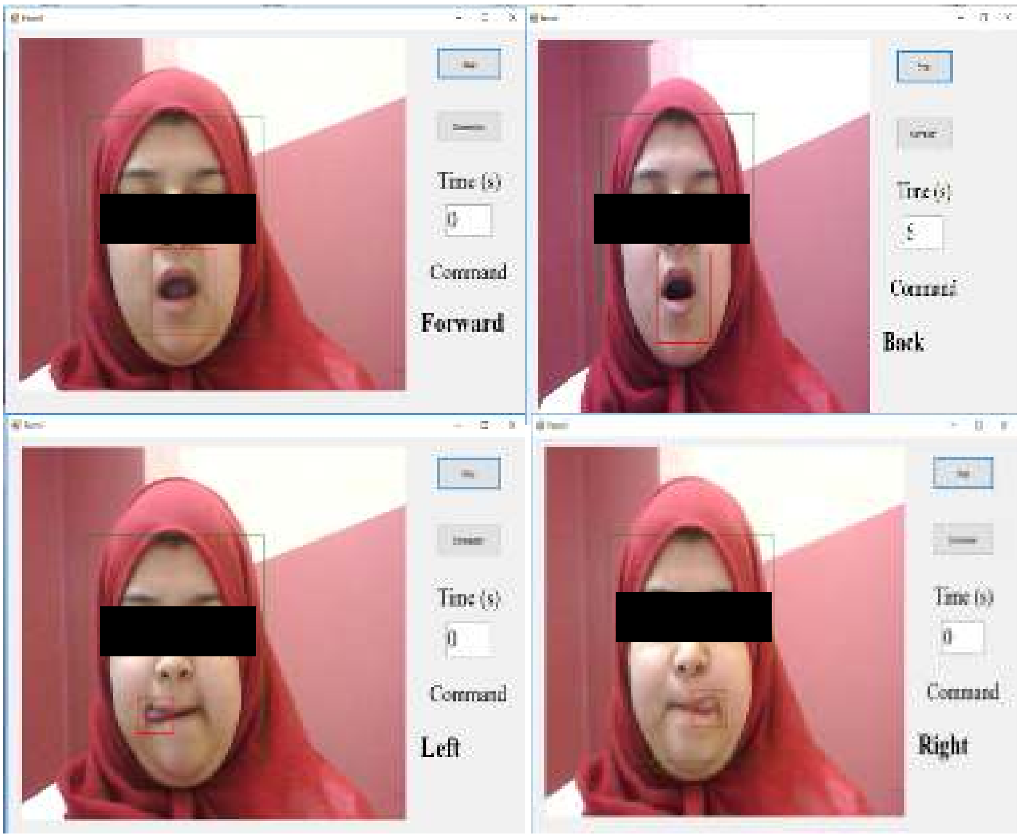


Figure 3.16 : Mouth gesture recognition results in a normal light condition.

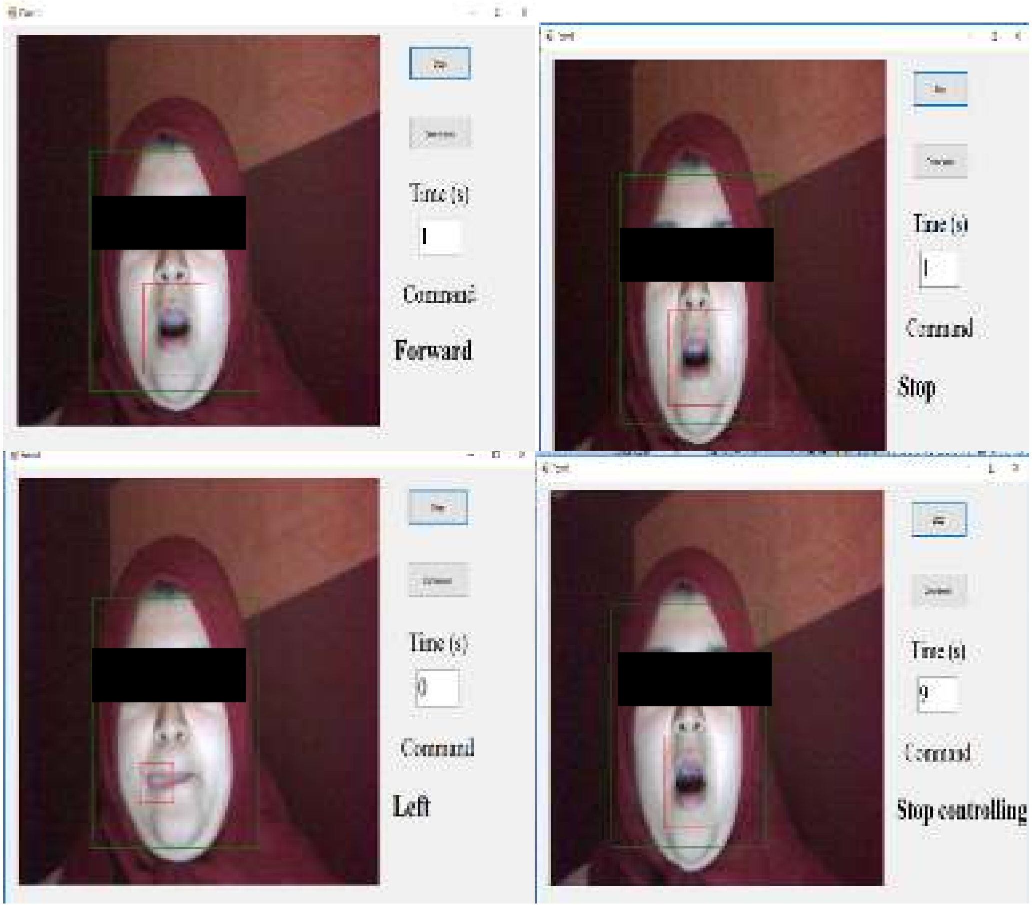


Figure 3.17 : Mouth gesture recognition results in a dark light condition.

As indicated in the figures above, the detected face is drawn in green color, and the recognized mouth gesture result is drawn in red color and written in command Label which is located in the right corner of the interface, the system accurately detected the face and mouth, confirming the robustness to time-varying illumination, and low sensitivity to a cluttered environment.

Table 3.2 presents the results of the system for the both user in a (bright, normal, and light condition).

	Bright light condition		Normal light condition		Dark light condition	
Command	**User 1**	**User 2**	**User 1**	**User 2**	**User 1**	**User 2**
Forward	20	19	20	20	19	19
Backward	19	19	19	20	19	19
Left	19	18	18	19	19	18
Right	19	19	19	19	19	18
Stop	20	20	20	20	20	20
Start controlling	20	20	20	20	20	20
Stop controlling	20	20	20	20	20	20
Total	**272**		**274**		**270**	
Accuracy	**97.14%**		**97.85%**		**96.42%**	

Table 3.2: The results of the system for the both user in different lighting conditions

From the results, we can find out that the system accuracy is satisfactory in the three lighting conditions.

3.5. Conclusion

In this Chapter, we present two Vision based interface systems for hands free control of an intelligent wheelchair. The first interface uses the head gesture and the second uses the mouth gesture to control the movement of the wheelchair. In both interfaces, the system is tested many times by two persons in scenes which had cluttered stationary background with a varying illumination. Experimental results show that the both system accuracy is quite satisfactory in the three lighting conditions. However, we found that the head gesture recognition system accuracy in bright and dark lighting condition is lower than in a normal lighting condition.

CHAPTER 4

SENSOR-BASED HUMAN MACHINE INTERFACE SYSTEM

4.1. Introduction

In this chapter, a sensor based HMI system for hands free control of an IW is presented. The developed system is based basically on the patient head gesture (head tilt movement). The head gesture is detected using accelerometer and gyroscope sensors embedded on a single board, both IMU(Inertial Measurement Unit) sensors output are combined together using Kalman filter as sensor fusion to build a high accurate orientation sensor. The system is tested many times. Experimental results show that the system accuracy is quite satisfactory.

4.2. Head motion control system

The system uses head motion recognition technique based on IMU sensors to control the wheelchair. The flowchart of head motion control system is shown in Figure 4.1[34].

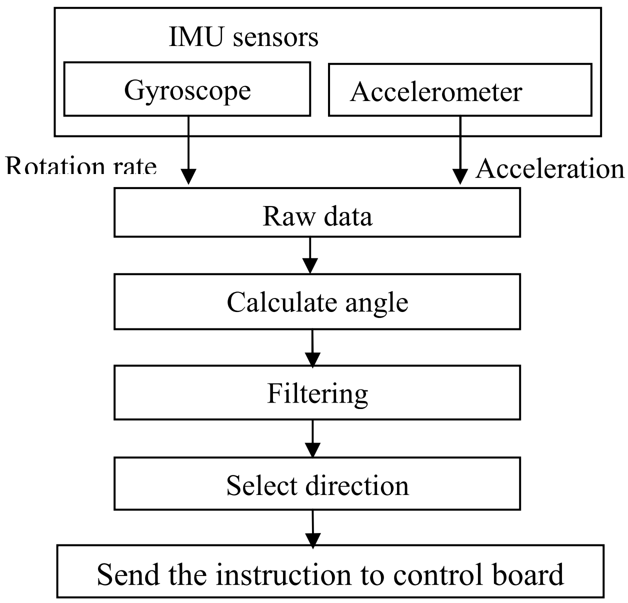


Figure.4.1: Flowchart of head motion control mode

As shown in Figure 4.1, the algorithm is implemented through several steps of operation. Firstly, the sensor board must be placed horizontally on the head tilt of the user, when the user moves his/her head in different directions around x and y axis the acceleration and rotation rate of the movement is detected as raw value. The head movement around Z-axis is not used to give the user the ability of moving his/her head around without affecting the control of the system (Figure 4.2). Then, the main board estimated the corresponding head angle using geometrical calculation and sensor fusion with Kalman filter. The estimated head angles are compared with angle thresholds to select the direction of the wheelchair.

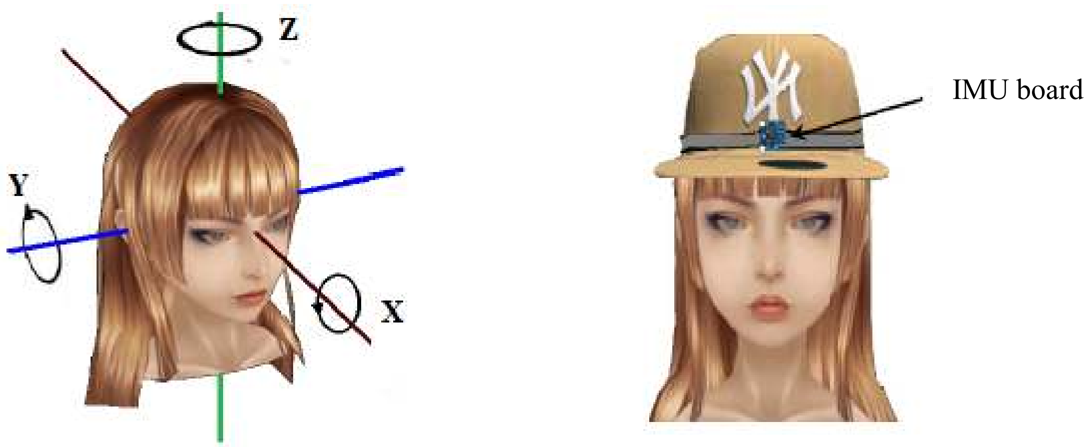


Figure 4.2: X, Y, and Z orientation axes and IMU board position

4.2.1. Head gesture estimation

The roll and pitch angles of the user's head are used to control the wheelchair. The comparison of their values with prefixed thresholds is used to send the exact instruction to the wheelchair control board. Therefore, it is important that the estimated head gesture angles must be as accurate as possible.

The sensor provides some raw value but its output is relatively noisy. MEMS (Micro-Electro-Mechanical System) gyroscopes use the Coriolis acceleration effect on vibrating mass to detect angular rotation. The gyroscope measures the angular velocity, which is linear to rate of rotation. It responds quickly and accurately, but it has a tendency to drift over time because it only senses changes and has no fixed reference frame. Therefore, the platform system using a gyroscope only, could not work in case of high precision requirements.

The output of the gyroscope is the angular rate of rotation about the three axes. Integrating the angular rotation rates over a time period yields angular rotation increments. Therefore, after summing all the increments, we can obtain the orientation estimation relative to the starting orientation. But if there was a small error during the integration, the small error is accumulated over time and provokes drift in the resulting orientation estimation.

On the other hand, while the gyroscope causes drift when orientation is estimated, the accelerometer is not affected. Combining the outputs of the two sensors can provide better estimation of orientation. The accelerometer measures linear acceleration based on the acceleration of gravity, it is more accurate in static calculation when the system is closer to its fixed reference point. The problem of accelerometer is that it tends to distort acceleration due to external forces as gravitational forces in motion, which accumulates as noise in the system and erroneous spikes in resulting output. The long-term accuracy of a gyroscope combined with the short-term accuracy of the accelerometer, these sensors can be combined to obtain more accurate orientation reading by utilizing the benefits of each sensor.

A few methods to apply sensor fusion are available to varying degrees of complexity. Kalman Filter is the most popular sensor fusion algorithm as it does not take a lot of processing power for a better behaving system.

In order to analyze data provided by the IMU sensor, the raw data needed to be converted into angular units.

Roll angle φ_a is obtained using the accelerometer value of a_x and a_z along x and z axes, while pitch angle θ_a is found using the accelerations a_y and a_z along y and z axes.

$$\varphi_a = atan2(a_x, a_z) \tag{4.1}$$

$$\theta_a = atan2(a_y, a_z) \tag{4.2}$$

Data for gyroscope are found similarly, except that the gyroscope output represents angular rates $\dot{\varphi}$ and $\dot{\theta}$ in deg/s.

$$\varphi_g = \dot{\varphi}_{g\_prev} + \dot{\varphi}_{g\_new}.\Delta t \tag{4.3}$$

$$\dot{\varphi}_g = \frac{\dot{\varphi}_g - \dot{\varphi}_{g\_bais}}{s} \tag{4.4}$$

$$\theta_g = \dot{\theta}_{g\_prev} + \dot{\theta}_{g\_new}.\Delta t \tag{4.5}$$

$$\dot{\theta}_g = \frac{\dot{\theta}_g - \dot{\theta}_{g\_bais}}{s} \tag{4.6}$$

Where $\dot{\varphi}_g$ is the roll rate, $\dot{\theta}_g$is pitch rate of the gyroscope, Δt is the time step, $\dot{\varphi}_{g\_bais}$ the offset calibration found on initialization and s is the sensitivity of the sensor found in specification sheet for the IMU .

4.2.2. Sensor fusion

Kalman Filter was designed by selecting pitch or roll angle as a state vector and using accelerometer to estimate gyroscope constant deviation b. The discrete system state and measurement equations with the deviation as a state vector are as follows:

$$\begin{cases} \hat{x}_k = A\,\hat{x}_{k-1} + B\,u_k \\ \hat{y} = C\,\hat{x}_k \end{cases} \tag{4.7}$$

Where $\hat{x}_k$ denotes states vector at k time, $\hat{y}_k$ accelerometer angular measurement values at k time, A the state space matrix, B control matrix, C observation matrix and u_k gyro output containing fixed drift at $k-1$ time.

The equation (4.7) can be expanded to (4.8) when using the equation (4.1) and equations from (4.3) to (4.6)

$$\begin{cases} \begin{bmatrix} \varphi_k \\ b_k \end{bmatrix} = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \varphi_{k-1} \\ b_{k-1} \end{bmatrix} + \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} \dot{\varphi}_{gk} \\ \hat{y}_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \varphi_{ak} \\ b_k \end{bmatrix} \end{cases} \tag{4.8}$$

Where φ represents the pitch angle, $\dot{\varphi}_{gk}$ represents gyroscope output, and φ_{ak}represents the pitch angle calculated from accelerometer.

The discrete-time Kalman Filter equations are as follows [69]:

Step prediction of the state:

$$\hat{x}_k^- = A\,\hat{x}_{k-1} + B\,u_k \tag{4.9}$$

State estimation is:

$$\hat{x}_k = \hat{x}_k^- + K(\hat{y}_k - C\,\hat{x}_k^-) \tag{4.10}$$

Where the superscript _ denotes prior estimate value and K is the Kalman gain.

By inserting equation (4.8) to (4.9) and (4.10) we can get the Kalman equations for the IMU sensor:

$$\hat{\varphi}_k^- = \varphi_{k-1} + (\dot{\varphi}_k - b_{k-1})\,\Delta t \quad (4.11)$$

$$\varphi_k = (1 - K_0)\varphi_k^- + K_0 a_k \quad (4.12)$$

$$b_{g_k} = b_{g_{k-1}} + K_1\left(\varphi_{a_k} - \varphi_{a_k}^-\right) \quad (4.13)$$

The error covariance matrix for prediction estimation is

$$P_k^{\,-} = A\,P_{k-1}A^T + Q \quad (4.14)$$

Where

P : is prediction uncertainly matrix,

Q: is the system process noise covariance matrix,

Kalman gain is:

$$K_k = \frac{P^-\,C^T}{C\,P^- C^T + R} \quad (4.15)$$

$$K_{k,00} = \frac{P_{00}^{\,-}}{P_{00}^{\,-} + R} \quad (4.16)$$

Where R is measurement noise covariance matrix from accelerometer, the updating error covariance prediction equations to:

$$P_{00} = \left(1 - K_{k,00}\right)P_{00}^{\,-} \quad (4.17)$$

The roll angle is estimated using the same process by choosing the roll angle as the state vector and calculating the prediction using Kalman filter.

4.2.3. Action selection

The direction of wheelchair depends on how the user moves his/her head as shown in Figure 4.3. Five commands are allowed by the control system: moving forward, moving backward, turning right, turning left and stop. If the user's head roll or pitch angle exceeds 20° than the wheelchair will move in the corresponding direction (Forward, backward, right, and left). If the user still or back to the center position, that make the wheelchair stopped. Threshold values can be adapted according to the user’s capacity.

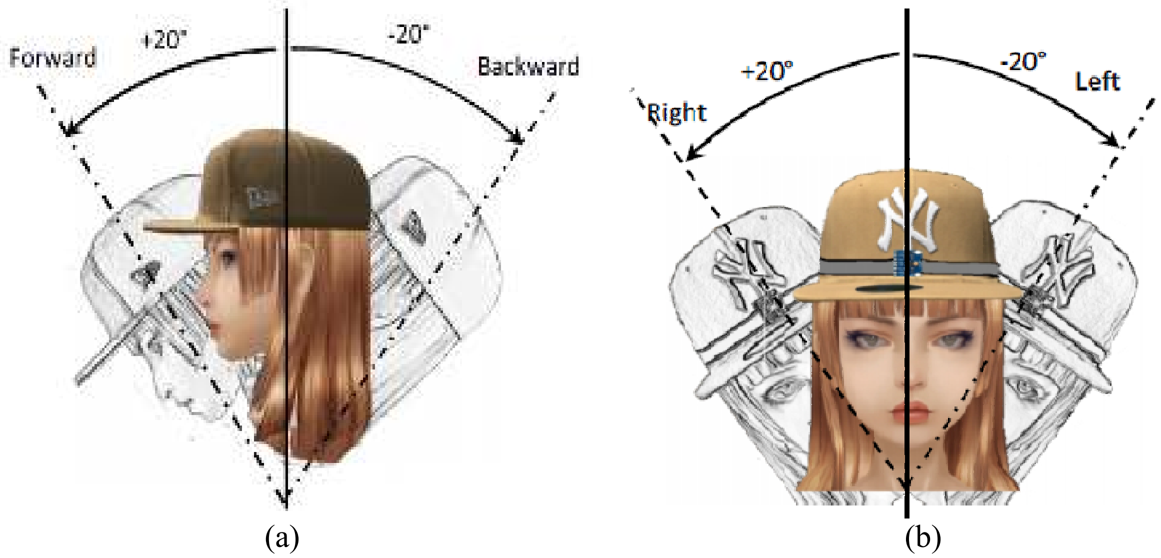


Figure 4.3: The position of the IMU board relative to head motion in different direction: (a) Left-Right, (b) Forward-Backward.

4.3. Results

In order to test the accuracy and the effectiveness of these interface two experiments were carried out. The first one is to evaluate the performance of the Kalman filter which was used to estimate the orientation of the IMU board. The second one is to evaluate the performance of our proposed system.

For the first experiment, back and forth movements around X-axis were applied to the IMU board. The roll angle was obtained by integrating the gyroscope output and then it was compared with the Kalman filter result. Figure 4.4 shows the comparison between the Kalman filter's output (blue line) and the gyroscope integration result (red line).

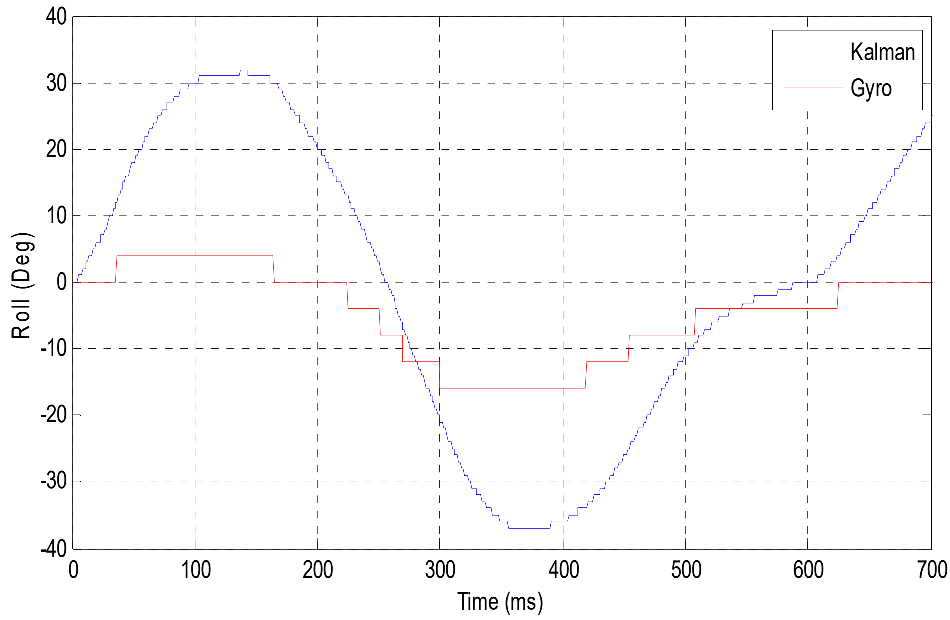


Figure 4.4: Comparison between the Kalman filter's output and the gyroscope integration result.

In this figure we can clearly see the big drift in the result of gyroscope integration, while this drift is eliminated in Kalman filter results.

Evaluation of Kalman filter's performance was continued by comparing the rotation angle from the accelerometer, which obtained by using equations (4.1) and (4.2), with the Kalman filter's output. Figure (4.5) shows the comparison between the Kalman filter's output (blue line) and the accelerometer result (red line).

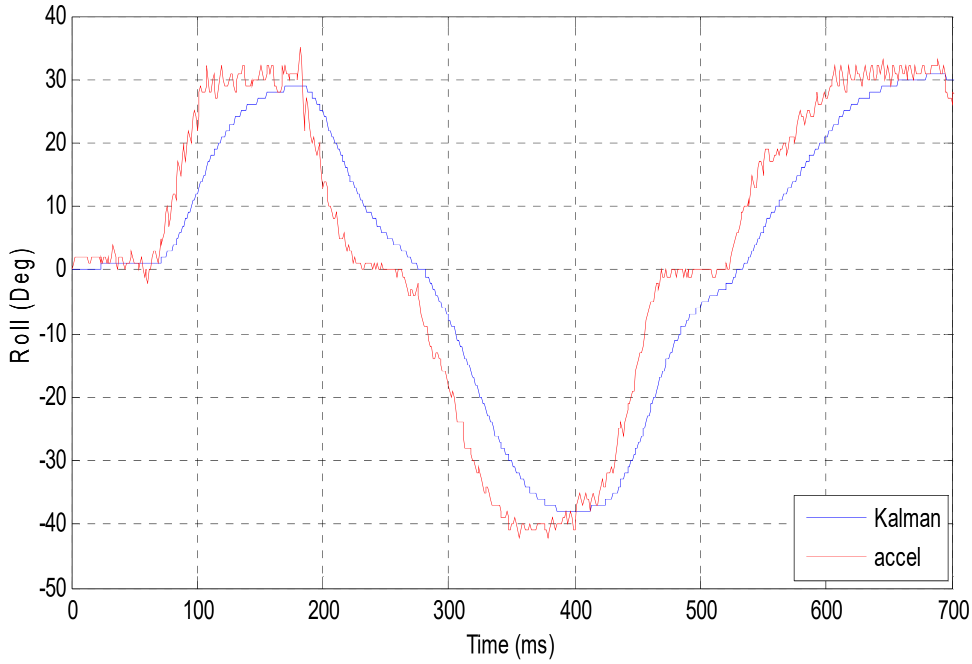


Figure 4.5: Comparison between Kalman filter output and the accelerometer result.

From the results, it is clearly observed that all fluctuation, which is seen in the accelerometer output, is eliminated successfully by the filter.

For the second experiment, firstly, the IMU board was placed horizontally on the front of a cap, worn by the user. Then, the five commands (Forward, Backward, Left, Right and Stop) were applied by one user thirty times for each command (fifteen times in experiment 1 and another fifteen in experiment 2). An example of issuing a series of commands is shown in Figure 4.6. The red lines indicate the thresholds chosen to select each action.

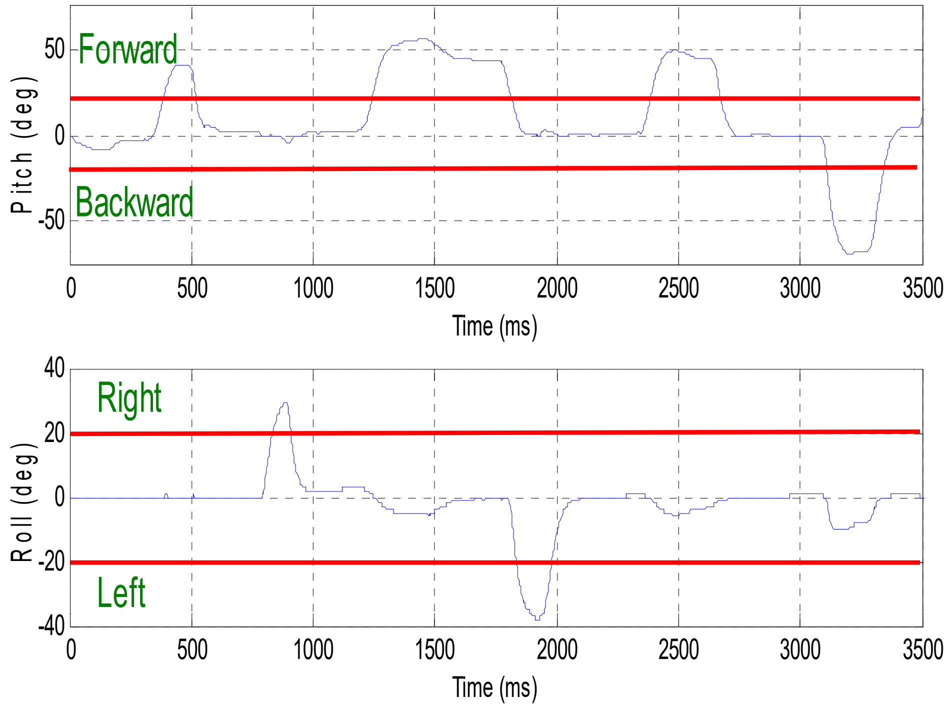
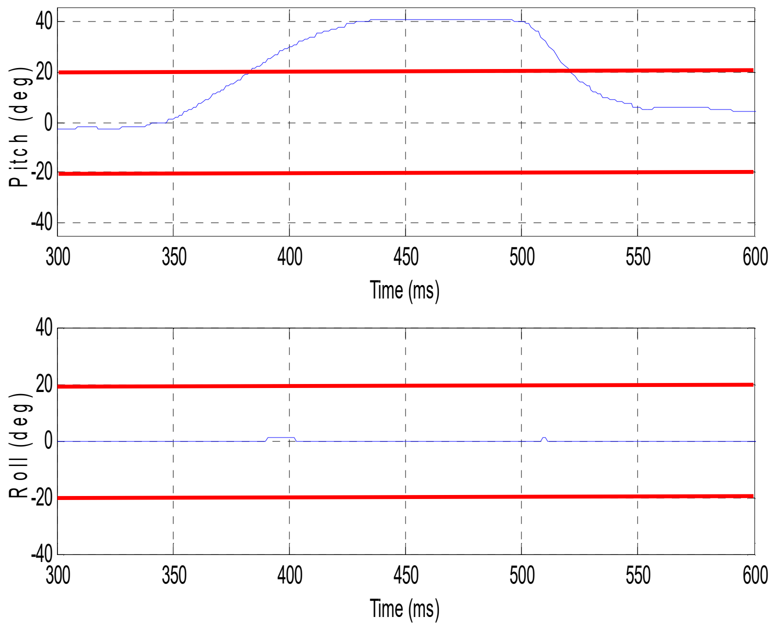
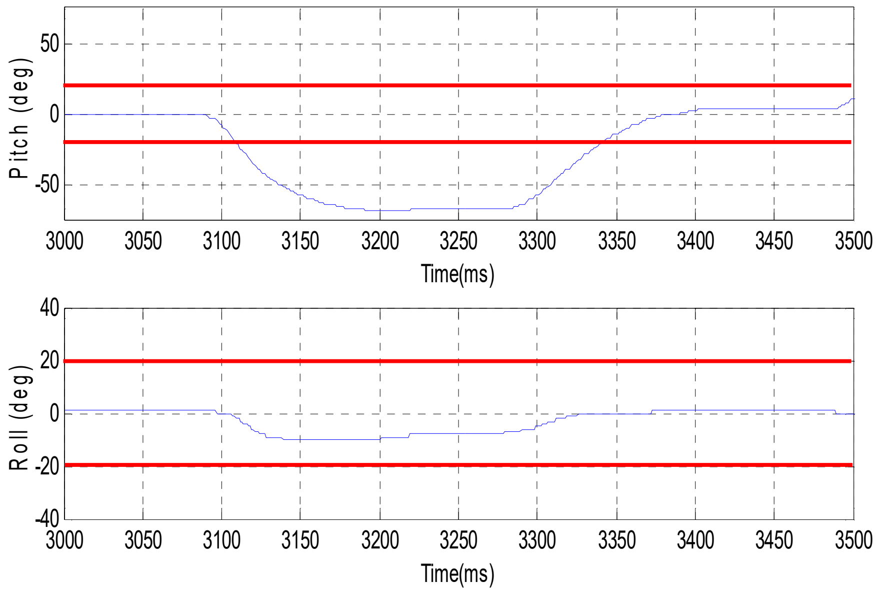


Figure 4.6: An example of issuing a series of commands.

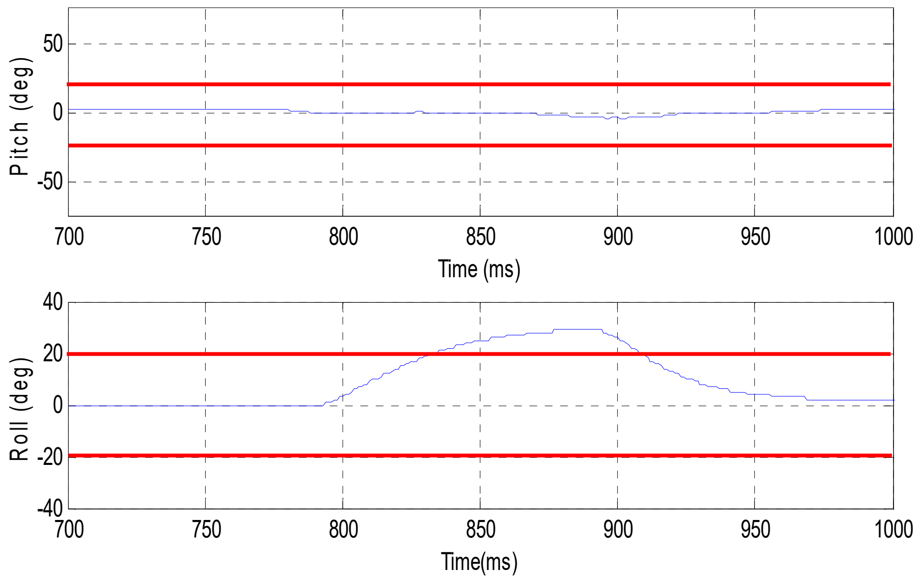
Figure 4.7 represents calculated angles (pitch and roll) when the user applies the five commands separately.



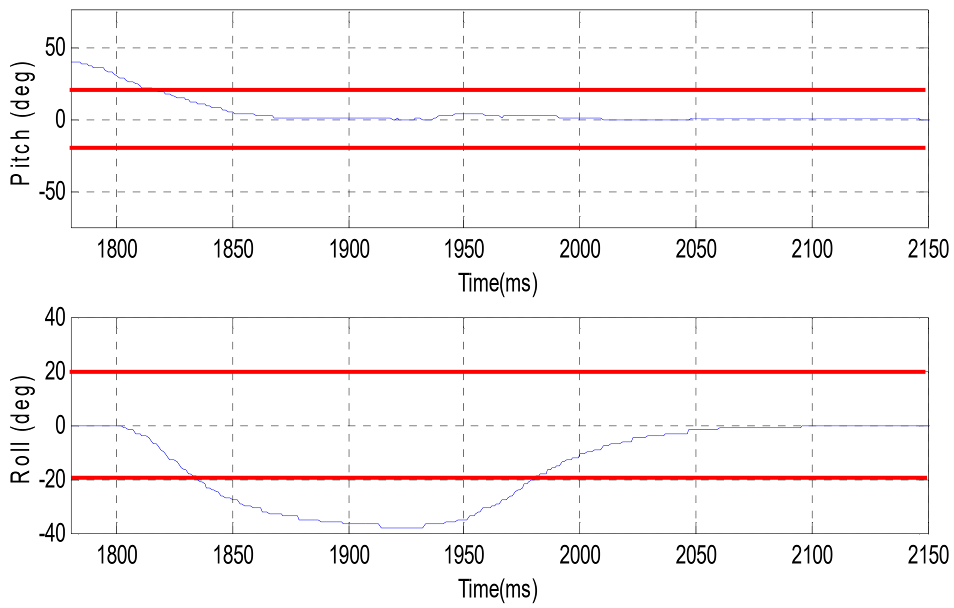
(a)


Pitch (deg)
50
0
-50
3000 3050 3100 3150 3200 3250 3300 3350 3400 3450 3500
Time(ms)
Roll (deg)
40
20
0
-20
-40
3000 3050 3100 3150 3200 3250 3300 3350 3400 3450 3500
Time(ms)

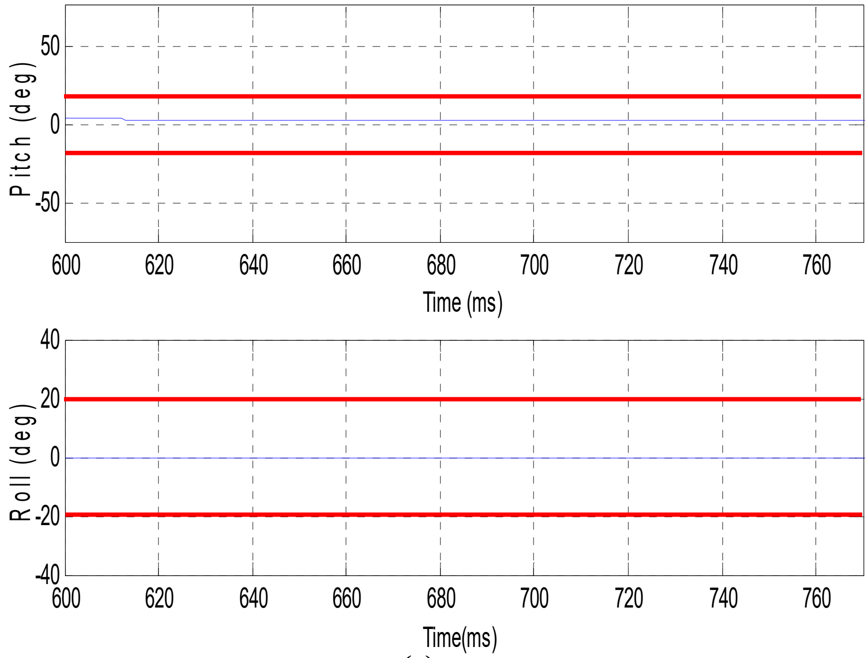
(b)


Pitch (deg)
50
0
-50
700 750 800 850 900 950 1000
Time (ms)
Roll (deg)
40
20
0
-20
-40
700 750 800 850 900 950 1000
Time(ms)

(c)


Pitch (deg)
50
0
-50
1800 1850 1900 1950 2000 2050 2100 2150
Time(ms)
Roll (deg)
40
20
0
-20
-40
1800 1850 1900 1950 2000 2050 2100 2150
Time(ms)

(d)



(e)

Figure 4.7: The calculated angles when the user applies the five commands separately.

In Figure 4.7 (a), when the user tilts his head down with an angle 20° or more, the gesture is recognized as the forward movement. The pitch angle must be equal or more than 20° and the roll angle must be more than -20° and less than 20°.

In Figure 4.7 (b), when the user tilts his head up with an angle 20° or more, the gesture is recognized as the backward movement. The pitch angle must be equal or less than -20° and the roll angle must be more than -20° and less than 20°.

In Figure 4.7 (c), when the user slopes his head right with an angle 20° or more, the gesture is recognized as the right turn. The pitch angle must be more than -20° and less than 20°and the roll angle must be equal or more than 20°.

In Figure 4.7 (d), when the user inclines his head left with an angle 20° or more, the gesture is recognized as the left turn. The pitch angle must be more than -20° and less than 20°and the roll angle must be equal or less than -20°.

In Figure 4.7 (e), when the user keeps or backs his head at the center, the gesture is recognized as stop. The pitch angle must be more than -20° and less than 20°and the roll angle must be more than -20° and less than 20°.

Table 4.1 presents the results of the system of the two experiments.

Commands	**Experiment 1**	**Experiment 2**
Forward	15	15
Backward	14	15
Left	15	15
Right	15	15
Stop	15	15
Total	149	
Accuracy	**99.33%**	

Table 4.1: The results of the system of the two experiments.

From the results, we can find out that the proposed system is reliable for controlling the wheelchair.

4.4. Conclusion

In this Chapter, a head gesture recognition system based on IMU sensors is designed to control an intelligent wheelchair. The sensor board fixed to the front of a cap worn by the user detects head orientation and sends data to control board to calculate the appropriate head inclination angles using geometry rules and sensor fusion using a Kalman filter to build a high accurate orientation sensor. Experimental results show that the proposed system is reliable for controlling the wheelchair.

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1. Introduction

This chapter presents the architecture of the proposed multimodal IW, most relevant implementation details and the results achieved by the implemented prototype.

The outline of this chapter is as follow: at first, we present the concept, design, the implementation of the platform and the different hardware and software used for the development of our IW. The second part contains the concept of the proposed multimodal interface, a description of its basic input methods and how the multimodal system works and the different tests applied to the wheelchair platform and their results.

5.2. Wheelchair platform

This section presents the concept, design, the implementation of the platform and the different hardware and software used for the development of our IW.

5.2.1. Synoptic of our electric wheelchair

Different criteria must be considered for choosing a wheelchair like: weight limit, size, outdoor or indoor use, seat, armrests, footrests, backrests, wheels, electric controls, batteries, motors, accessories.

The wheelchair used in this project is shown in Figure 5.1.

Size (H x W x D)	127 x 70 x 106 cm
Seat dimension (H x W x D)	45 x 49 x 62 cm
Front wheels Size	20 x 5 cm
Back wheels Size	37 x 7 cm
Battery size	22.7 x 13.4 x21 cm
Batteries weight	13 kg
Net weight without batteries	74 kg
Charge capacity	200 kg
Engine power	2x450W/ 24V
Battery life	40 km
Max forward speed	9 km/h
Max back speed	6 km/h
Tilt capacity	12°
Battery charger	AC100-240V50/60HZ 2A Output : 24V-4A
Charging time	8 h
Electronic	Joystick Module with actuator control. VR2

Figure 5.1: Technical characteristics of our electric wheelchair

Each mobile platform consists of a mechanical part and an electrical part. The mechanical part consists of the metal structure and the actuators. The electrical part includes the sensors, the electric motors, the power board and the control board. The electrical parts of the wheelchair are shown in the following synoptic (Figure 5.2).

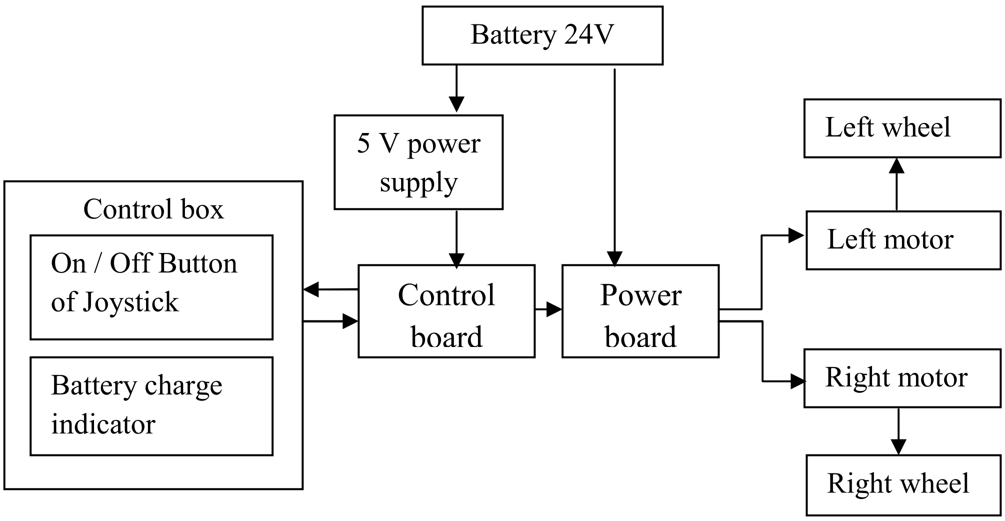


Figure 5.2: Architecture of the electrical part of the wheelchair.

5.2.2. Hardware Description

The hardware architecture of our system consists of two basic parts namely transmitter unit and receiver unit. The bloc diagram of these parts is shown in figure 5.3 and figure 5.4 respectively.

The transmitter side contains a laptop with a webcam, and a microphone and connected via serial port to an Arduino Uno card with an IMU sensor(MPU6050) and a Bluetooth module(HC05). . Whereas the receiver unit consists of a control board and an Arduino Mega card with a second Bluetooth module (HC06) and a belt of ultrasonic sensors.

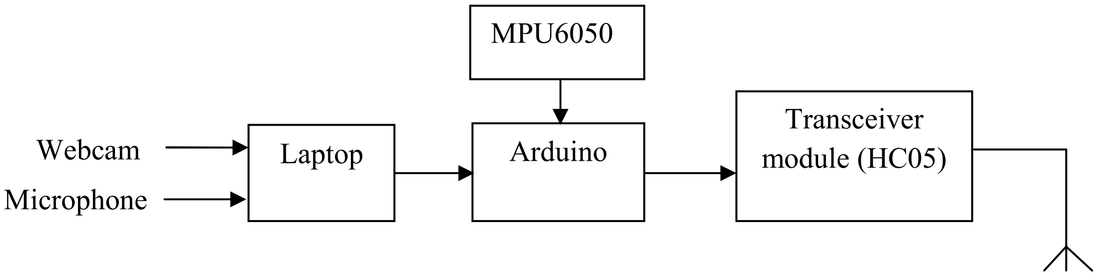


Figure 5.3: Bloc diagram of transmitter unit

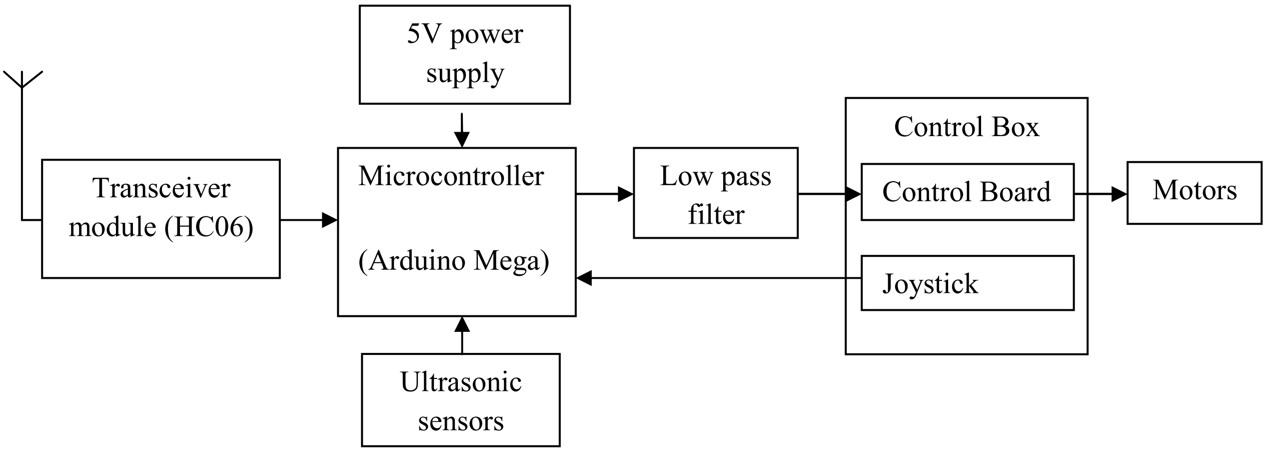


Figure 5.4: Bloc diagram of receiver unit.

Figure 5.5 shows an image of the IW platform. It contains the following components:

Figure 5.5: IW platform

5.2.2.1. Laptop

To run the platform software used, a laptop is used, it is equipped with headset microphone and HD Webcam. However, other computers with equivalent or superior performance may be used.

5.2.2.2. MPU6050 IMU board

The MPU-6050, shown in figure 5.6 is the world's first motion tracking device designed for the low power, low cost, and high performance requirements of smart phones, tablets and wearable sensors. The MPU-6050 device comprises a combination of a 3-axis gyroscope and a 3-axis accelerometer on the same silicon die, together with an onboard Digital Motion Processor (DMP) for motion fusion and a peripheral controller.

The MPU 6050 main features are [37] :

- Programmable accelerometer full-scale range: ±2g, ±4g, ±8g, and ±16g,
- Programmable gyroscope full-scale range: ±250 °/s, ±4g, ±8g, and ±16g. ±250, ±500, ±1000, and ±2000 °/sec,
- Programmable output data rate: max: 1 kHz,
- I2C and SPI communication interface,
- Internal Digital Motion Processing unit.

The MPU6050 is used to measure the wheelchair patient's head gesture and send data to the microcontroller for processing. Estimated head gesture, using geometry rules and sensor fusion, is used to control the wheelchair movement.

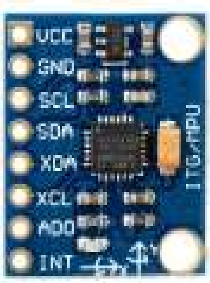


Figure.5.6: MPU6050 IMU board

5.2.2.3. Ultrasonic sensors

In order to maintain a safer distance from obstacles, a set of ultrasonic sensor modules are placed around the wheelchair. The ultrasonic sensor used here is HC-SR04 [29] .

Figure.5.7: HC-SR04 module.

This sensor is compact, arduino compatible; and provides precise and stable non contact distance measurement from about 2 cm to 400 cm with very high accuracy. The module includes ultrasonic transmitter, receiver and control circuit.

5.2.2.4. Control and data acquisition boards:

The boards are used to collect data from the sensors and send instructions to the wheelchair control board. The developped system is based on two boards(Arduino Mega and Arduino Uno). The first is connected to the computer platform via a USB, its role is to send the command to the second arduino via Bluetooth module, while the second is connected to control board and its role is to receive the commands from the first arduino and send the instruction to the wheelchair control board.

The Arduino Mega 2560 [71] is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports). The Arduino UNO[72] is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs. The both boards have a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. They contain everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

(a) arduino Mega 2560 (b) arduino UNO

Figure.5.8: Arduino boards.

5.2.2.5. Bluetooth modules (Transceiver modules)

HC Serial Bluetooth product consists of Bluetooth serial interface module and Bluetooth adapter. Bluetooth serial module is used for converting serial port to Bluetooth. Bluetooth serial module's operation doesn't need drive, and can communicate with the other Bluetooth device. But communication between two Bluetooth modules require at two conditions: i) The communication must be between master and slave. ii) The password must be correct. In this project we used two Bluetooth modules; the first one is HC05 as master [28], and the second HC06 as slave [23].

- HC-05 is a most capable module that can be set to be either Master or Slave.
- HC-06 is a Slave only device. (It looks physically just like the HC-05).
- These small (3 cm long) modules run on 3.3V power with 3.3V signal levels, They have no pins and usually solder to a larger board. The module has two modes of operation, Command Mode where we can send AT commands to it and Data Mode where it transmits and receives data to another bluetooth module.
- "Breakout Boards» are available and recommended. These mount the sub-module like that shown on the right on a slightly larger board.

Figure.5.9: Bluetooth modules.

5.2.2.6. Control Box

In order to avoid any technical damage to the electric wheelchair, we tried to make a small change to control box by designing a virtual joystick using an arduino which delivers to the wheelchair the same signals coming from X and Y-axis of joystick if the joystick is active, and emulates joystick signals according to command instructions become from transmitter part, if the joystick is inactive then, one from the other modes is used.

a. Working principles of joystick

The joystick, JC 2000 contactless joystick from Curtis-Wright, is a two-axis potentiometer with two outputs per axis as shown in Figure 5.10 [21].

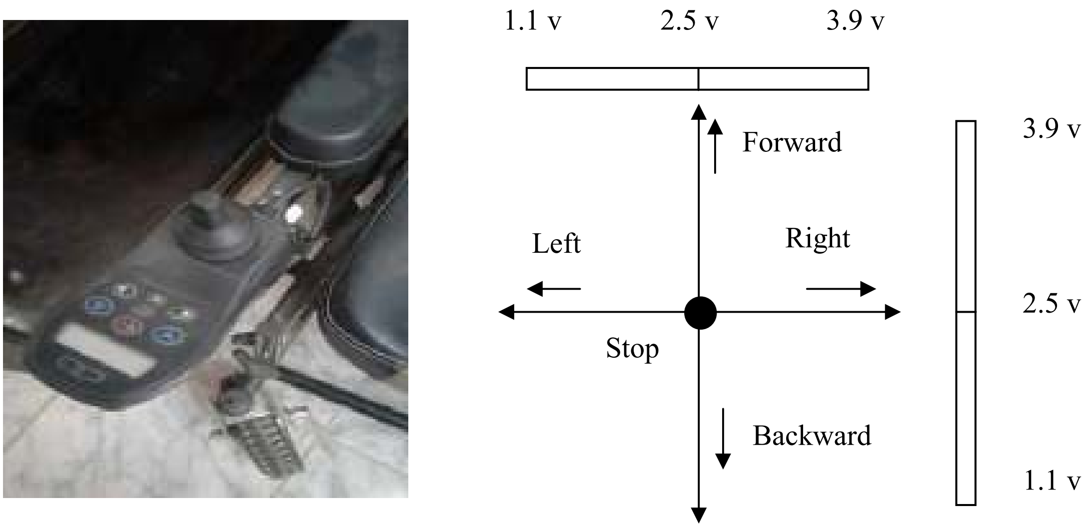


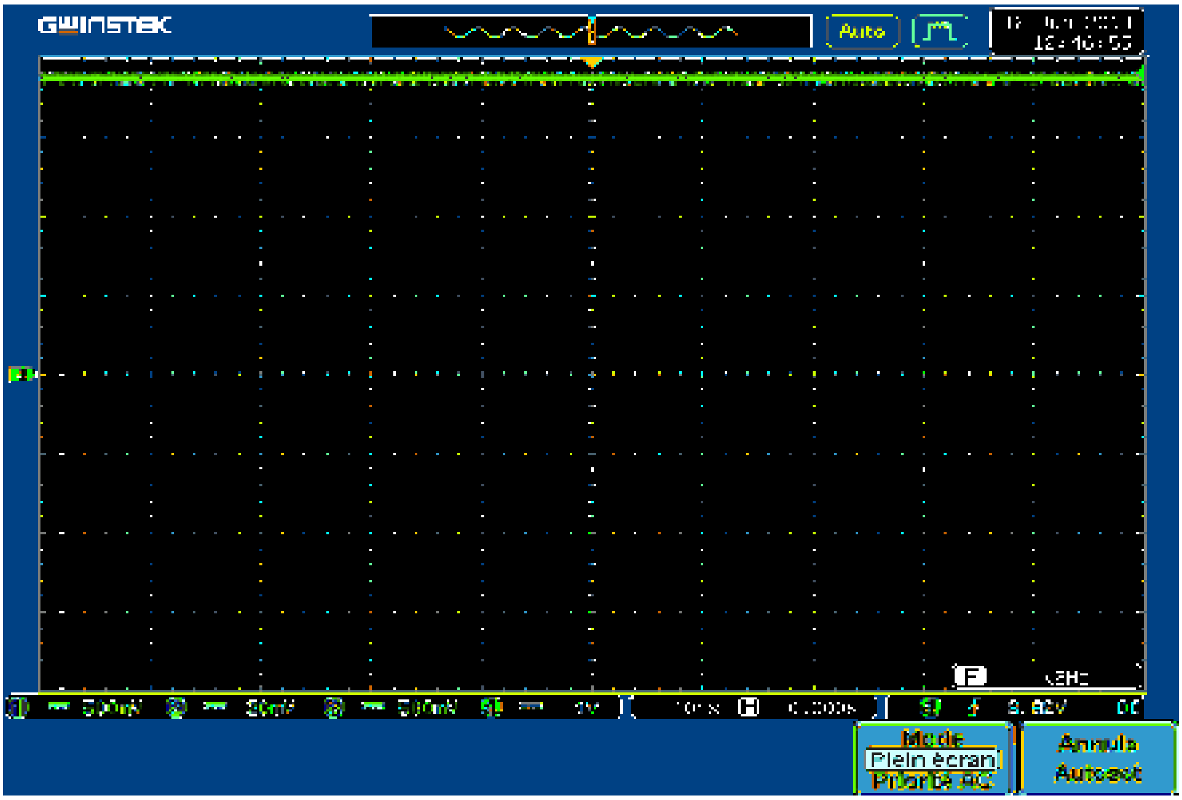
Figure 5.10: The Joystick and the generated voltage from joystick axis.

The voltages generated by the axis movement are used to control the wheelchair movement via VR2 controller. If the joystick moves along X axis then the wheelchair turns right or left, if the joystick moves along Y axis then the wheelchair goes forward or backward. Table 5.1 shows the joystick voltage ranges.

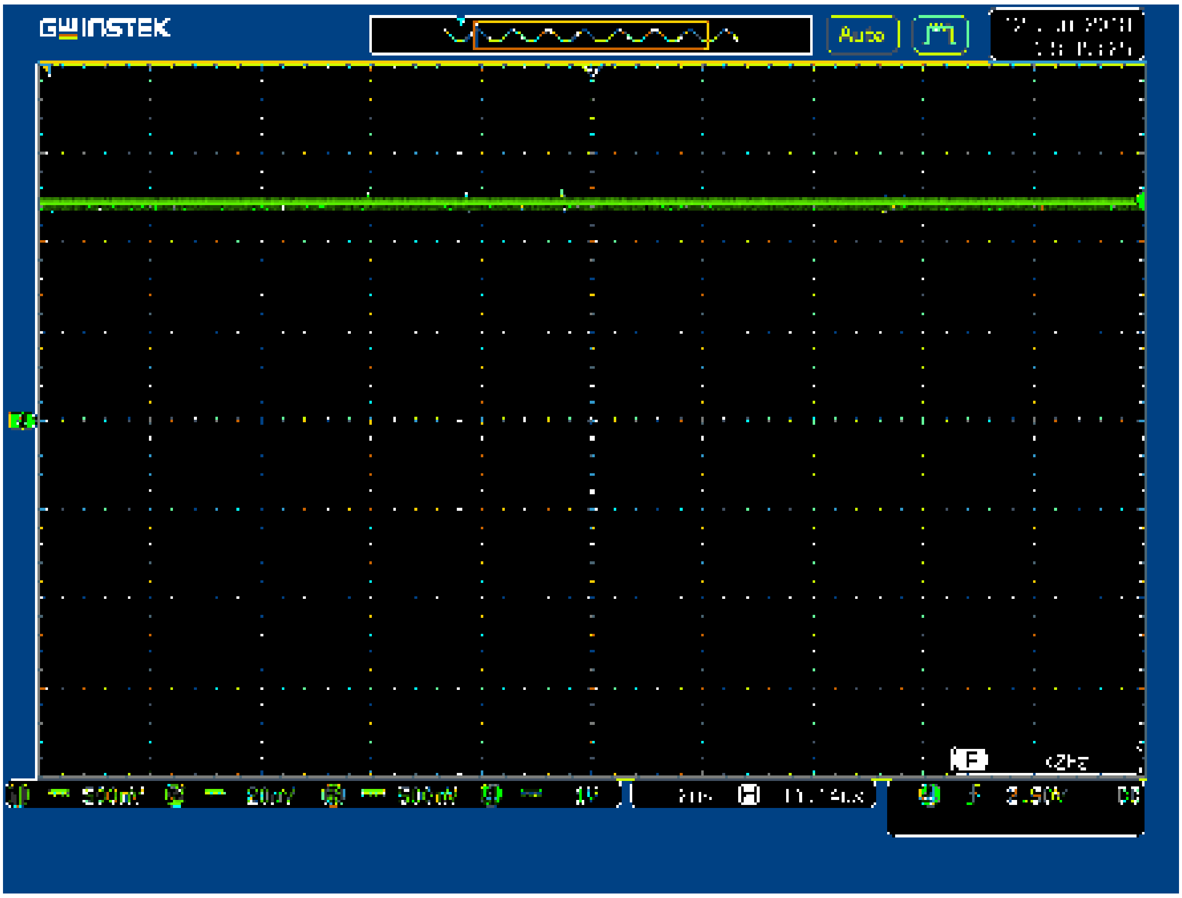
	Output1	**Output2**
Stop	2.5 V	2.5 V
Forward	2.5 V	2.5 V~3.9 V
Backward	2.5 V	1.1 V~2.5 V
Turn right	2.5 V~3.9 V	2.5 V
Turn left	1.1 V~2.5 V	2.5 V

Table 5.1: The joystick voltage ranges.

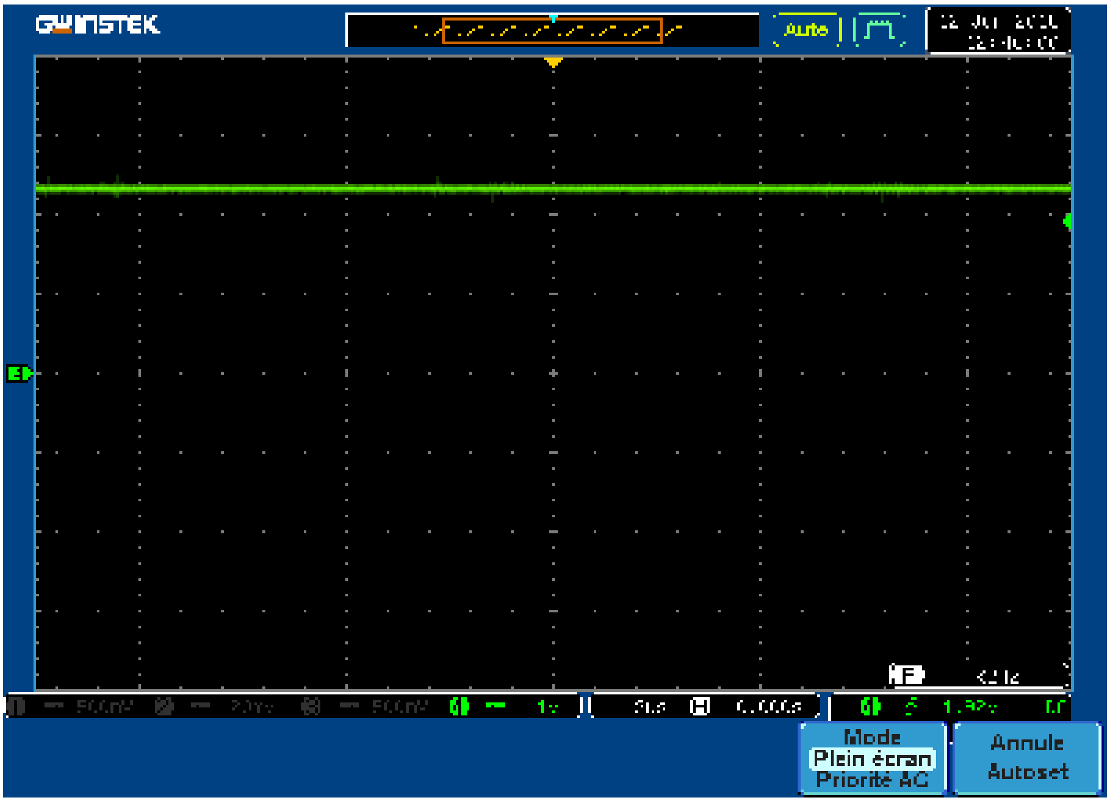
Figure 5.11 show continuous voltages delivered by the joystick when the potentiometer is oriented forward, it is reinstated to the center, and then it is oriented backward.



(a) Forward



(b) Center



(c) Backward

Figure 5.11: Voltages delivered by the joystick

b. Control board conception

In order to make the microcontroller board emulates joystick signals, we used its Pulse Width Modulation (PWM) outputs [3]. Then, we converted it to voltage using RC low pass filter. Figure 5.12 shows low pass filter circuit which chosen to convert PWM output signal become from arduino to analog voltage with a minimal ripple and fast response.

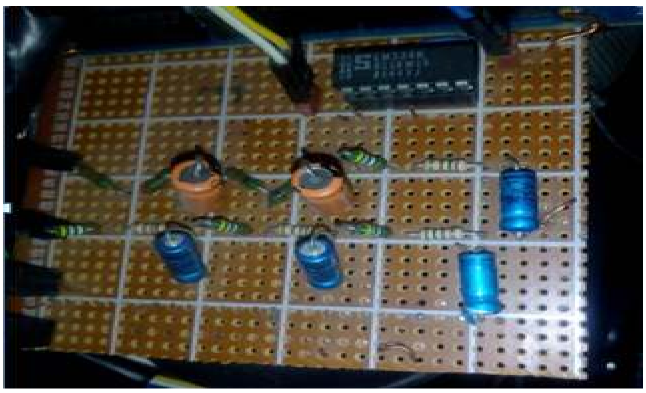
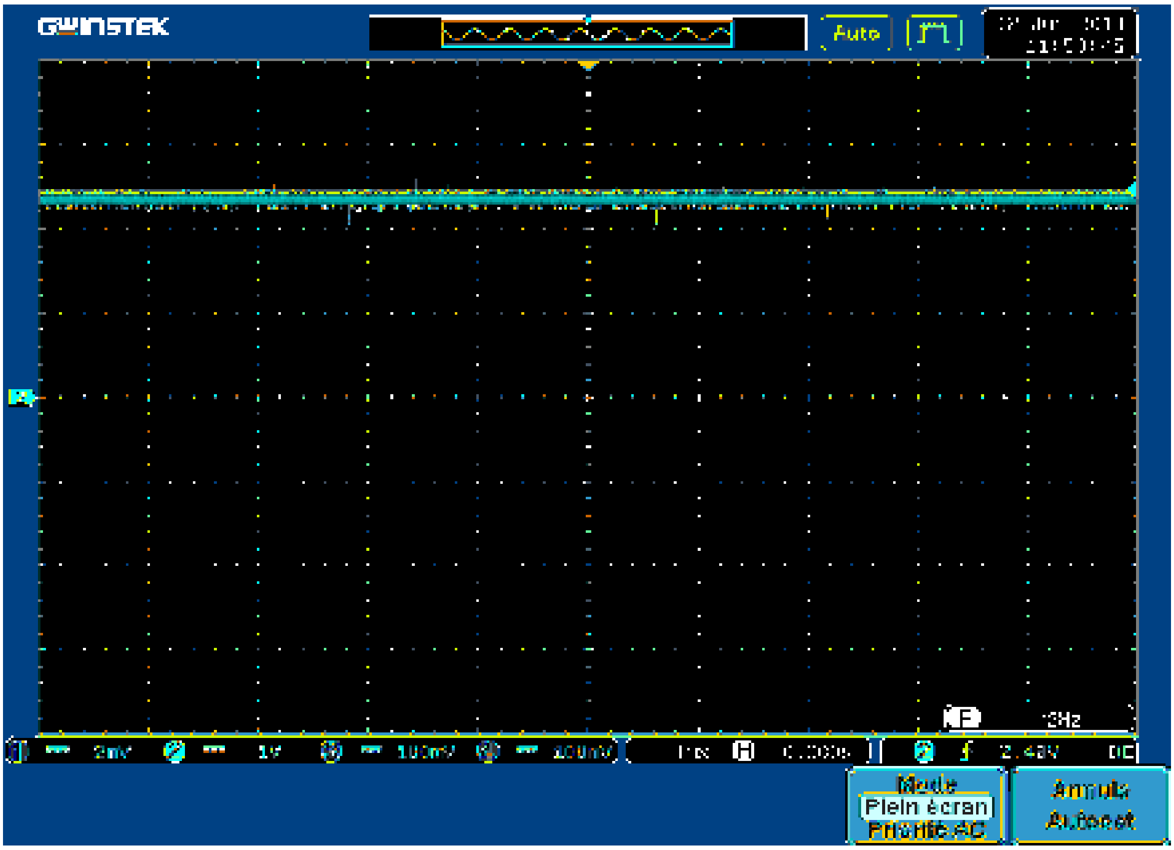
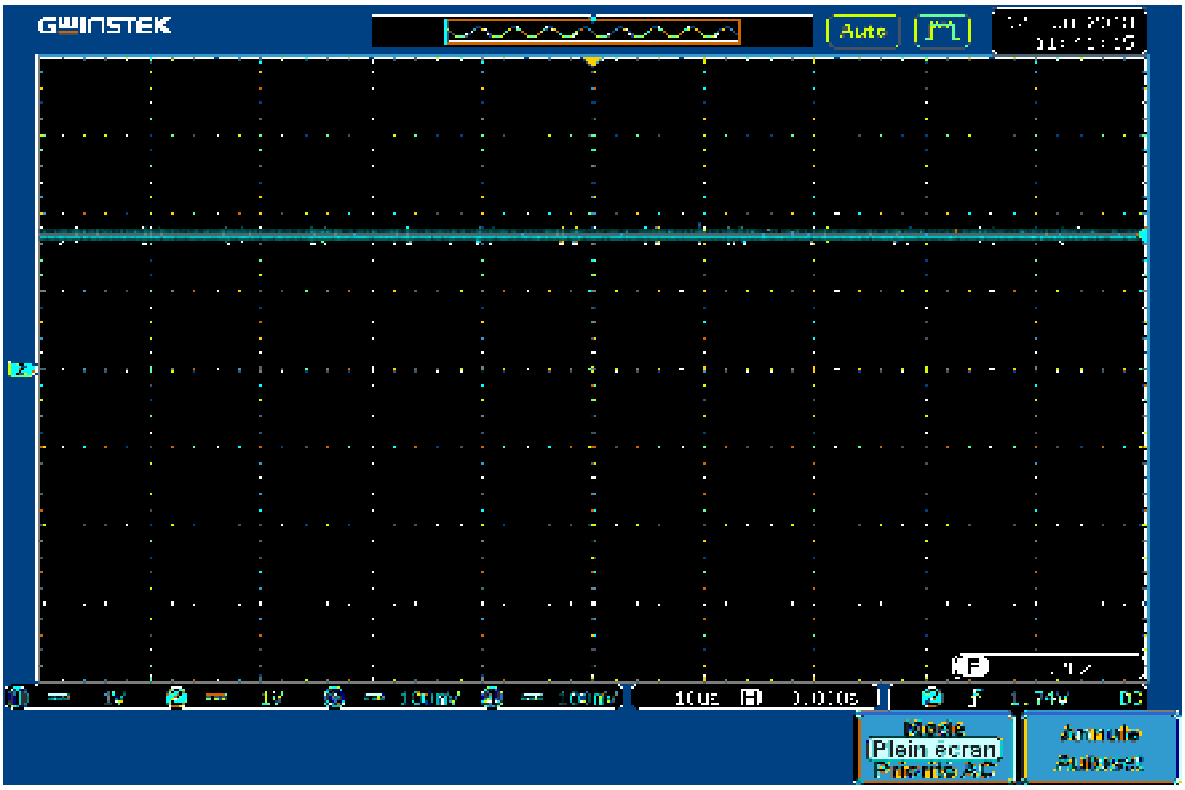


Figure 5.12: RC low pass filter circuit

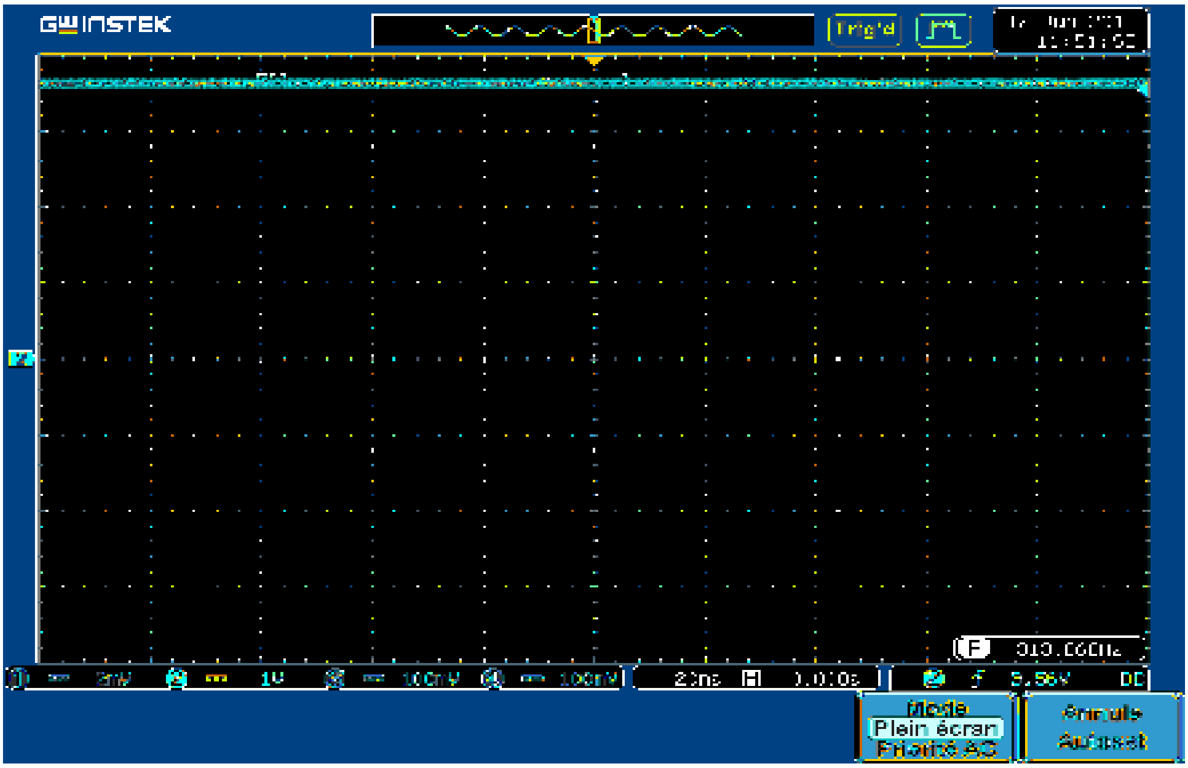
Figure 5.13 presents voltages delivered by the control board.



(a) Stop



(b) backward



(c) Forward

Figure 5.13: Voltages delivered by the control board.

5.2.3. Software used

The system uses Microsoft visual C#, EmguCV, Microsoft windows speech recognition and arduino-1.0 as a software program environment.

- **Microsoft Visual C#:** is an object-oriented programming language from Microsoft that aims to combine the computing power of C++ with the programming ease of Visual Basic. C# is based on C++ and contains features similar to those of Java [26, 64].
- **The Microsoft Windows Speech Recognition**: is the speech recognition system that comes built into Windows Vista, Windows 7, Windows 8, and Windows 10 include version 8.0 of the Microsoft speech recognition engine. Speech Recognition is available only in English, French, Spanish, German, Japanese, Simplified Chinese, and Traditional Chinese. Windows 7 that enables users to control the mouse cursor and keyboard through speech recognition [52].
- **EmguCV library**: Emgu CV[24] is a cross platform .Net wrapper to the OpenCV image processing library. Allowing OpenCV functions to be called from .NET compatible languages such as C#, VB, VC++, IronPython etc. The wrapper can be

compiled by Visual Studio,Xamarin Studio and Unity, it can run on Windows, Linux, Mac OS X, iOS, Android and Windows Phone.

- **Arduino IDE**: The Arduino IDE [5] is an open source programming environment based on Processing and is used to program the Atmega328 on the Arduino boards. The Intelligent Gesture Controlled Wireless Wheelchair For The Physically Handicapped. IDE has several inbuilt functions and falls under AVR embedded C design based on Processing.

5.2.4. Obstacle detection

In order to maintain a safer distance from obstacles, three modules of ultrasonic sensors are placed at the bottom of the wheelchair, in such away, that the area between 2-3 m in the front is covered. The arduino read the information from the sensors, if there is any obstacle the wheelchair will be stopped.

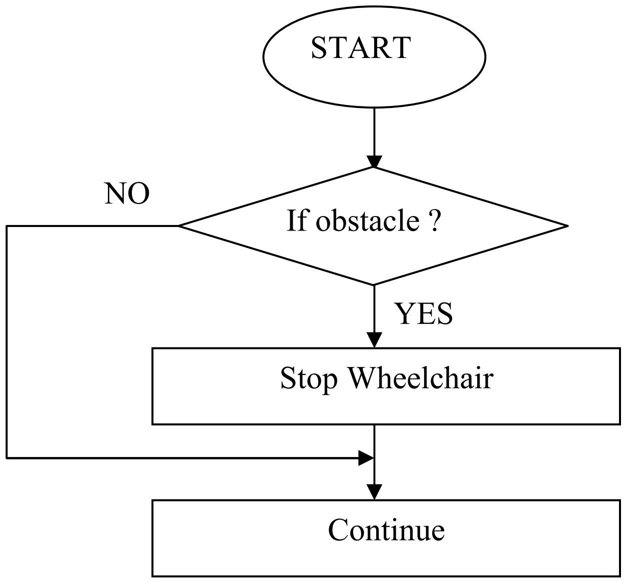


Figure 5.14: Flowchart Representing the Working of the obstacle Sensor

5.3. System implementation

This section presents the concept of the proposed multimodal interface, a description of its basic input methods and how the multimodal system works. As well as different applied tests to wheelchair platform and their results.

5.3.1. Multimodal control system

The proposed multimodal interface offers four basic control modes (Manual, vision based method, sensor based method and speech based method). In order to control the movement of the wheelchair, firstly, the vision mode which uses mouth gesture recognition to control the IW is selected automatically. If the user's health condition allows him/her to move his/her head or to use his/her voice. Then, he can select one from the other modes by stopping the first mode to active head gesture mode which uses IMU sensor or by saying "demarrer le système" (Start the system) to active speech recognition mode and stop the vision mode. In order to avoid any disagreement in multimodal control of the wheelchair, the joystick has higher priority than the other modes to use it in emergency case.

Once the control mode is selected, the controlling command is determined by the joystick, mouth gesture, head motion, or speech. Then, according to these commands, the microcontroller generates signals to emulate the joystick for controlling the power circuits of the wheelchair. Figure 5.15 shows the structure of multimodal control system.

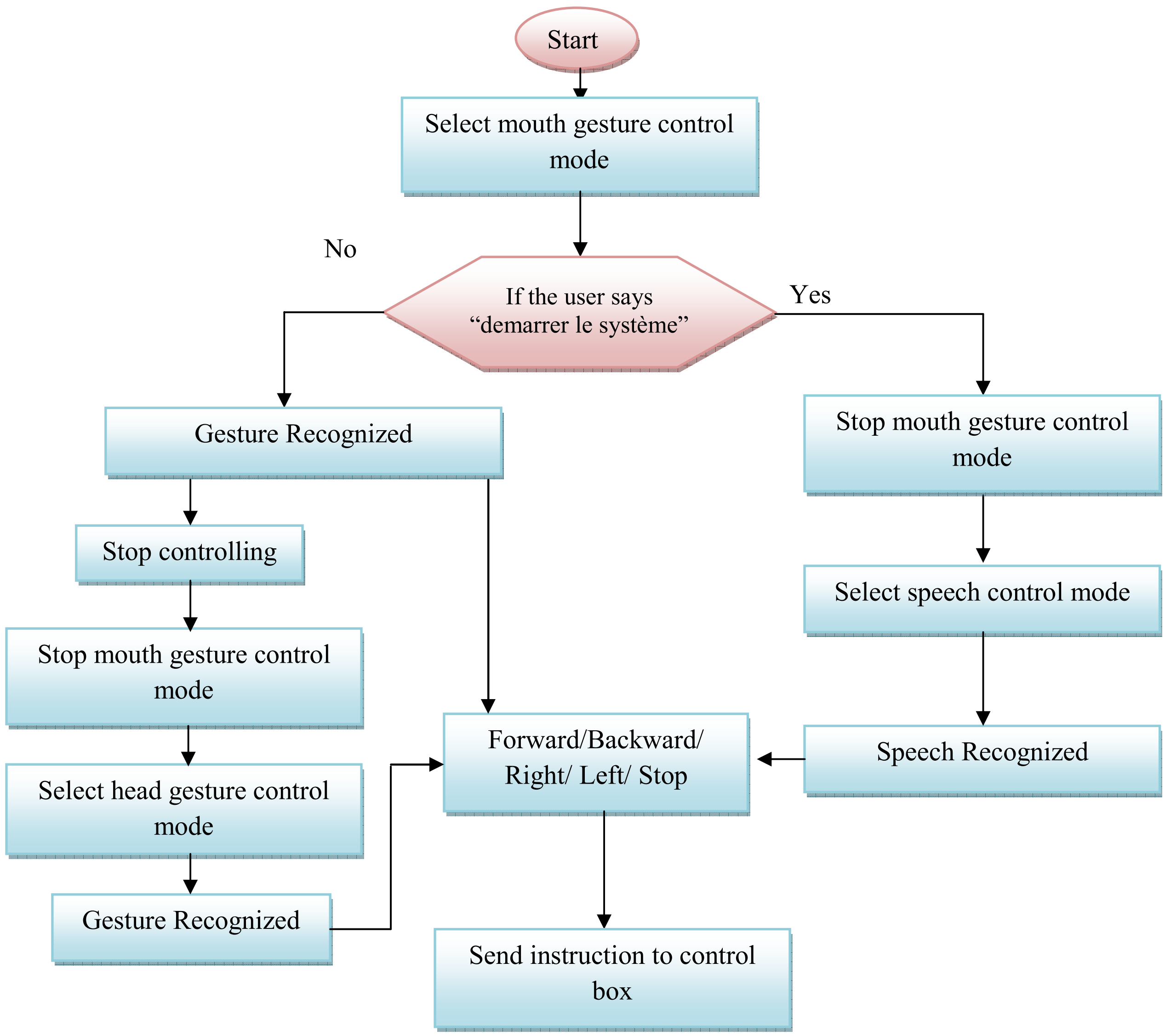


Figure 5.15: The structure of multimodal control system.

5.3.2. User Interface

Figure 5.16 shows the user interface, the left side is windows showing real time image from the camera, Capture/Stop button used to start or to stop capturing image in real time, Connexion button used to start sending the instruction to the wheelchair via Serial Port, Time Label presents the time taken when executing open mouth motion, command Label shows the gestures recognition results, and control mode Label shows the control mode chosen by the user.

Figure 5.16: The user interface

5.3.3. Experimental Results and analysis

The experiment consists of two steps. In the first step, two healthy students were asked to learn how the system works by at first selecting one from the three proposed control mode three times for each mode. Then, the selection is followed directly by repeating the five commands (Forward, backward, left, right and stop) ten times. Thus, 150 commands for each mode are tested.

In the second step of the experiment, they preformed the four control modes of the wheelchair (mouth gesture, head motion, speech and manual mode) in laboratory environment.

In order to apply the five commands, we choose to follow the route presented in Figure 5.17. As can be seen, The width of the wheelchair is about 60 cm , the distance between desks is 2 m , and total distance is about 9 m .

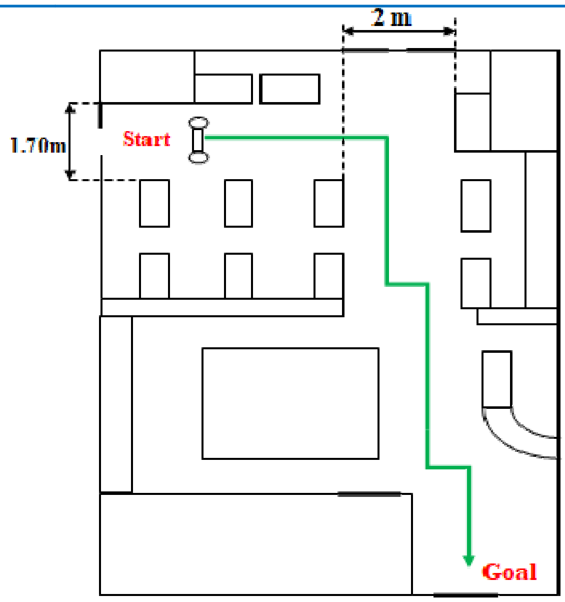


Figure 5.17: The route to follow in the laboratory environment

Two tests were conducted in this part. In each one, the traveling route is recorded and the number of input commands used to track the trajectory is calculated. Detailed results are given in Table 5.2 and Table 5.3.

	Control mode	Test	Number of wrong commands				
			Forward	Backward	Right	Left	Stop
User 1	Mouth gesture	T1	4	3	2	2	3
		T2	2	1	2	1	1
		T3	1	1	0	1	0
	Head motion	T1	0	2	1	0	0
		T2	0	1	0	1	0
		T3	0	0	0	0	0
	speech	T1	1	2	1	0	2
		T2	0	1	0	0	0
		T3	1	1	0	0	1
User 2	Mouth gesture	T1	3	2	1	2	3
		T2	2	0	1	0	3
		T3	2	1	0	1	2
	Head motion	T1	1	3	2	1	0
		T2	0	1	0	1	1
		T3	0	0	0	0	0
	speech	T1	2	3	1	1	3
		T2	0	1	0	0	1
		T3	1	0	0	0	2

Table 5.2: Result of the first part of the experiment.

			Number of input commands	Number of wrong commands
User 1	Manual mode	T1	22	0
		T2	20	0
	Mouth gesture mode	T1	28	3
		T2	26	2
	Head motion mode	T1	24	0
		T2	20	0
	Speech mode	T1	24	2
		T2	22	1
User 2	Manual mode	T1	20	0
		T2	20	0
	Mouth gesture mode	T1	30	4
		T2	26	2
	Head motion mode	T1	26	1
		T2	22	0
	Speech mode	T1	28	2
		T2	24	1

Table 5.3: Result of the second part of the experiment.

Figure 5.18, and figure 5.19 shows the time for the both users following the route for two times using four different control modes.

Figure 5.21, Figure 5.22, Figure 5.23 present the sequence of images that our IW is entering using the mouth gesture mode, head motion mode, and speech mode respectively.

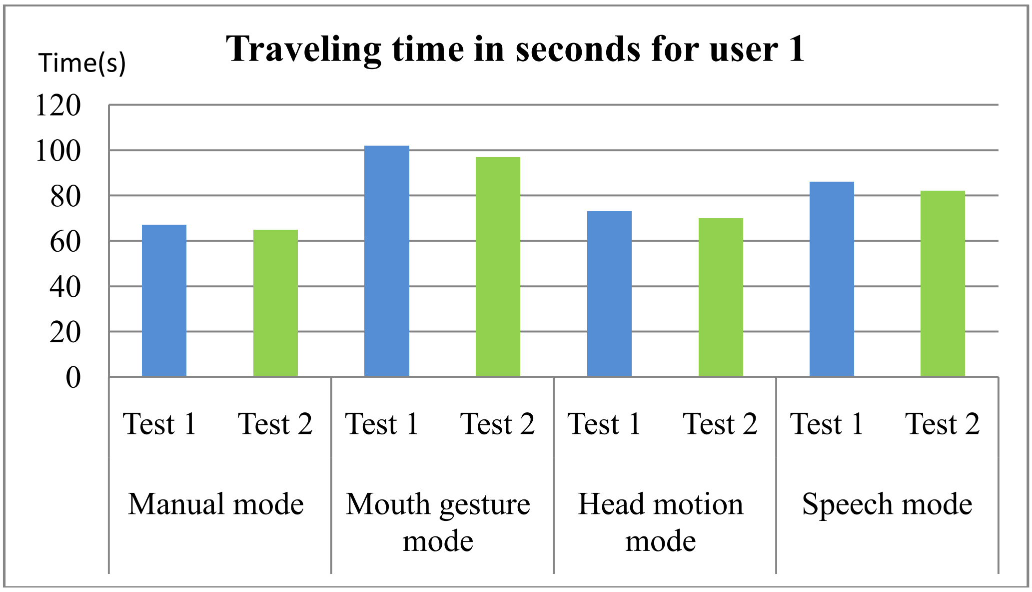


Figure 5.18: Time in seconds for user 1 following the route for two times using four different control modes.

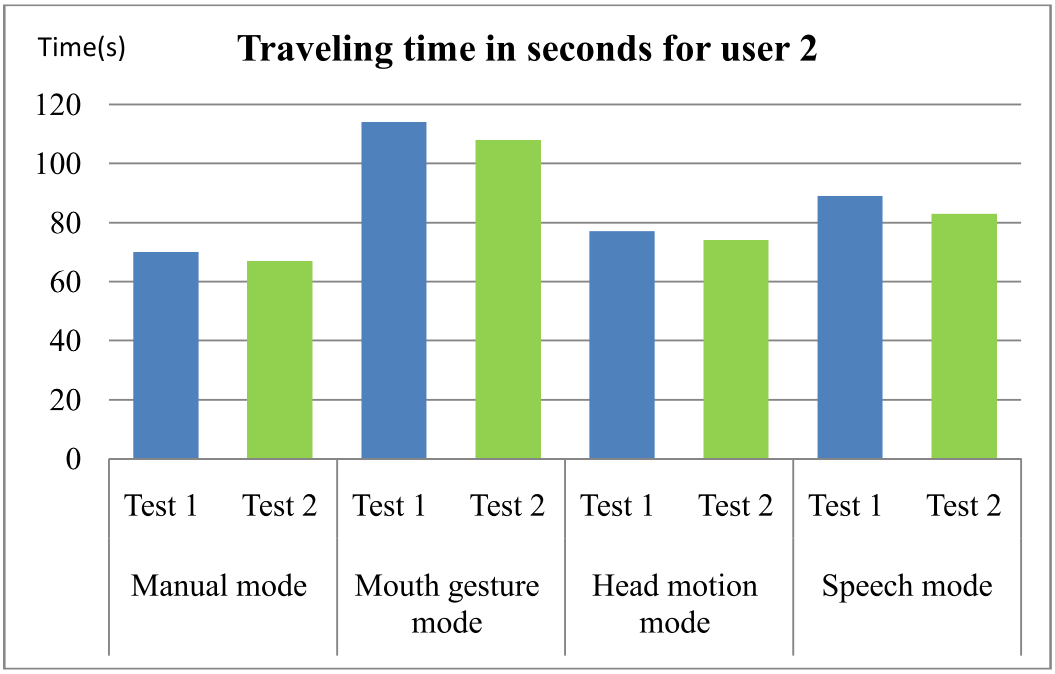


Figure 5.19: Time in seconds for user 2 following the route for two times using four different control modes.

Figure 5.20: Both user who performed the experiment

Figure 5.21: Experiments of mouth gesture mode.

Figure 5.22: Experiments of head motion mode.

Figure 5.23: Experiments of speech mode.

As can be observed in Table 5.2 and Table 5.3, most of the errors that emerge in the first part of the experiment decrease with the test 1 and test 2 while issuing a mouth gesture control mode.

Also, the errors that emerge in the second part of the experiment decrease with the test 2 while performing mouth gesture and speech control mode.

The reason for the test 1 errors is that the user had not trained well to how performs the commands.

Figure 5.18 shows the time used by user 1 to travel the route two times per control mode. As can be seen, the fastest control mode was the manual mode, it took 67 seconds and 65 seconds to finish the route the second fastest control modes was head gesture, it is slightly slower than manual mode, they had almost similar time 73 second and 70 seconds. The third fastest control mode was speech control, finishing the route in 86 seconds and 82 seconds. Finally, the mouth gesture control mode is the slowest control mode; it took more than 90 seconds to finish the route in both tests.

Figure 5.19 shows the time achieved by user 2 to travel the route two times per each control mode. As in the case of user, the manual mode is the fastest control mode followed by head gesture, speech and mouth gesture control mode, respectively.

The reason for the slowness of mouth gesture and speech control mode is that: some of mouth gesture control mode commands take a long time. For example, for executing backward command, the user must open his mouth more than 4 seconds. Where, the small delay of executing speech control mode commands caused by the pronunciation time of each command.

5.4. Conclusion

This chapter is divided into two parts: in the first part, the concept, design, and implementation of the platform. As well as Different hardware and software used for the development of our IW were presented. The second part was dedicated to describe how the multimodal system works and to show the experiments that were made during the implementation stage of the IW system. The results were very satisfactory. However, a number of improvements still need to be made to compass the desired level of quality.

CONCLUSION

The goal of this thesis was to develop and to design an IW with a multimodal interface which combines all the possible input methods and using necessary electronics to ensure safe mobility and ease of operation for different kinds of disabled persons. In order to achieve the proposed goal, we divided our work into four main steps.

In the first step, a study of some IW projects and their necessary requirements is done to understand their concept. Then, the most popular modes of HMI are revised to see the existing input devices that can be used in IW systems and recognition methods liable to be used by any kind of disabled people; this step was summarized in the first chapter.

In the second step, several HMI are applied to the electric wheelchair and presented in three different chapters depending on their recognition methods.

Chapter 2 presents HMI based on speech recognition using two different ways, the first way uses Microsoft Windows speech Recognition. The other way is using an android application in Smartphone which uses Google Speech to Text to recognize and process human speech. In both interfaces, the system is tested many times by two persons in silent and noisy environments. Experimental results show that the system accuracies are quite satisfactory in both environments.

In chapter 3, two vision based interface systems are presented. The first interface uses head gesture as an input method to control the movement of the wheelchair. Firstly, the head is detected using haar cascade. Once, the initial tracking window is determined, the new head location is determined using Camshift algorithm. Then, by checking the location of the center of the rectangle containing the user's head against reference rectangles, the head gesture commands are determined correspondingly. The second interface is based on mouth gesture recognition. This system consists of two main parts: the first part is mouth detection using haar cascade to find the area of the face and template matching to detect the mouth gesture from the lower face area. The second part is the command extraction, where, the controlling command is determined according the detected gesture and the corresponding reference template. In both interfaces, the system is tested many times by two persons. The exprimental results show that both interfaces can achieve the purpose of controlling the intelligent wheelchair.

In chapter 4, a sensor based HMI system for hands free control of an IW is presented. The developed system works basically on patient head gesture. The head gesture is detected using accelerometer and gyroscope sensors embedded on a single board. both IMU sensors output are combined together using Kalman filter as sensor fusion to build a high accurate orientation sensor.

In the next step, the characteristics of our wheelchair and the hardware chosen to implement the platform of the IW is identified and summarized in chapter 5, and followed by presentation of the concept of the proposed multimodal interface, a description of its basic input methods and how the multimodal system works. As well as different applied tests to wheelchair platform and their results. .

The experimental test shows the success of proposed multimodal control system. However, one may say that in order to take full advantage of the system, a training session is advised.

Despite having a good system specification and a functional prototype that implements most of the proposed input methods, there are still further steps that should be followed to obtain a more complete system for assisting people with different levels of disabilities:

- Develop a robust voice based HMI that could be embedded in our IW system.
- Extend the list of the input methods which embedded in the multimodal interface.
- Enhance the obstacle detection and avoiding algorithm.
- Study the recent developments in brain computer interfaces. Although it is under research, but it’s foreseen that one day it might break any physical disability barrier.

REFERENCES

[1] AdvancerTechnologies, Three-lead Differential muscle/ Electromyography Sensor for Microcontroller Applications. 2013.

[2] M.H. Alsibai,"A Study on Smart Wheelchair Systems", *International Journal of Engineering Technology and Sciences,* vol.4, no.1, December 2015.

[3] D.M. Alter, Using PWM Output as a Digital-to-Analog Converter on a TMS320F280x Digital Signal Controller , Texas Instruments, 2008.

[4] K.Anoop, R. Sathish , R.Nivasan , C. Pandian," Designing and Modeling of Voice Controlled Wheel Chair Incorporated with Home Automation", *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 3, no.2, pp. 53-60, April 2014 .

[5] Arduino.cc, arduino IDE, Available at: https://www.arduino.cc/en/Main/OldSoftware Releases#1.0.x , Accessed 24 June 2018.

[6] Aruna, C.D.Parameswari, A. Malini, M. G. Gopu,." Voice recognition and touch screen control based wheel chair for paraplegic persons", IEEE International Conference on Green Computing Communication and Electrical Engineering, pp. 1-8 , 2014

[7] L. Berger, *Traitement d'images et de vidéos avec OpenCV 3 en C++ (Windows, Linux, Raspberry)* , D-BookeR, 2017.

[8] R. Bhardwaj, P. Gupta, P. Jadhav , B. Kadam , A. Kedari, " Android Based Automated Smart WheelChair", *International Journal of Innovative Research in Computer and Communication Engineering*, vol.4, no.3, 2016

[9] G. Bourhis, K. Moumen, P. Pino, S. Rohmer, and A. Pruski. "Assisted navigation for a powered wheelchair". *Proceedings of IEEE Systems Man and Cybernetics Conference - SMC*, vol.3, pp. 553-558, 1993.

[10] G. R. Bradski, "Computer video face tracking for use in a perceptual user interface" *Intel Technology Journal*, pp.705–740, 1998.

[11] G. Bradski, and A. Kaehler, *Learning OpenCV Computer Vision with the OpenCV Library*, O'Reilly Media, Inc, 2008.

[12] S. Brahmbhatt, *Practical OpenCV*, Apress, 2013.

[13] R. Brunelli ,*Template Matching Techniques in Computer Vision: Theory and Practice*, Chichester, U.K: Wiley, 2009.

[14] J.A.E. Cabello, A. Spalanzani, C. Laugier "Multimodal Control of a Robotic Wheelchair: Using Contextual Information for Usability Improvement", *IEEE/RSJ International Conference on Intelligent Robots and Systems,* Nov 2013, Tokyo, Japan. 2013.

[15] J. Cannan, H. Hu, "*Human-Machine Interaction (HMI): A Survey*", *School of Computer Science and Electronic Engineering*, University of Essex, March 2013.

[16] T. Carlson, J. Del R. Millan," Brain–Controlled Wheelchairs: A Robotic Architecture", *IEEE Robotics and Automation Magazine,* vol. 20, no.1, pp. 65 – 73, March 2013.

[17] J.K. Chacko, D. Oommen, K. K. Mathew, N. Sunny, N. Babu," Microcontroller Based EOG Guided Wheelchair", *International Journal of Medical, Health, Biomedical, Bioengineering and Pharmaceutical Engineering*, vol. 7, no.11, pp.423-426, 2013 .

[18] Y. Cheng, "Mean shift, mode seeking, and clustering," IEEE Trans. Pattern Anal. Machine Intell., vol. 17, no.8, pp. 790-799, 1995.

[19] S. Cockrell, G. Lee, and W. Newman. " Determining navigability of terrain using point cloud data". In Procedding. *IEEE International. Conference Rehabil. Robot*., pp.1–6, Seattle Washington, Jun. 2013.

[20] D. Comaniciu, P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Trans. Pattern Analysis Machine Intell*, vol. 24, no. 5, 603-619, 2002.

[21] Curtis-Wright , JC2000 Multi Axis Contactless Joystick, available at: www.cw-industrialgroup.com/getattachment/a6bb9db0-f4be-47c9-8e9d-e8931694969f/jc2000_brochure ,accessed 23 June 2018.

[22] E. Demeester, A. Huntemann, J. Del, R. Millan, and H. Van-brussel, "Bayesian plan recognition for Brain-Computer Interfaces", *2009 IEEE International Conference on Robotics and Automation*, pp.653-658, 2009.

[23] disciplines, HC-06 Bluetooth module, available at: https://disciplines.ac-toulouse.fr/sii/sites/sii/files/ressources/didacticiels/programmation/arduino/bluetooth/arduino-parametragehc06.pdf , accessed 20 june 2018.

[24] Emgu.CV, available at www.emgu.com , accessed 24 June 2018.

[25] D. Exner, E. Bruns, D. Kurz, A. Grundhofer, and O. Bimber, "Fast and robust CAMShift tracking", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),* pp.9-16, 2010.

[26] A. Freeman**,** *Introducing Visual C# 2010*, Apress**,** Berkeley, CA, 2010.

[27] G.Facciolo, N. Limare, E.Meinhardt, "Integral Images for Block Matching", IPOL Journal , vol. 4 , pp. 344–369 , 2014.

[28] Gotronic, HC-05 Bluetooth module, available at: https://www.gotronic.fr/pj2-guide-de-mise-en-marche-du-module-bluetooth-hc-1546.pdf, accessed 20 june 2018.

[29] Gotronic, HC-SR04 ultrasonic sensor, available at: https://www.gotronic.fr/pj2-hc-sr04-utilisation-avec-picaxe-1343.pdf , accessed 20 june 2018.

[30] R.GRASSE, "*Aide à la navigation pour les personnes handicapées : reconnaissance de trajets* ", PhD thesis, lorraine university, France, 2007

[31] A.Gupta, N. Joshi, N. Chaturvedi, S. Sharma, V. Pandar, "Wheelchair Control by Head Motion Using Accelerometer*", International Journal of Electrical and Electronics Research*, vol. 4, no.1, pp.158-161, 2016.

[32] M. Hamzi, M. Boumehraz, R. Hassani, "Hand Gesture Based Wheelchair Control", *Second International Conference on Electrical Engineering ICEEB'18*, Biskra, Algeria, 2018.

[33] R. Hassani, M. Boumehraz, Z. Habba,"An Intelligent Wheelchair Control System based on Head Gesture Recognition", *International Electrical and Computer Engineering Conference*, Setif, Algeria, 2015.

[34] R. Hassani, M. Boumehraz, M. Hamzi, "Head Gesture Based Wheelchair Control", *Second International Conference on Electrical Engineering ICEEB'18*, Biskra, Algeria, 2018.

[35] R. Hassani, M. Boumehraz, M. Hamzi, Z. Habba, "Gyro-Accelerometer based Control of an Intelligent Wheelchair", *Journal of Applied Engineering Science and Technology*, vol.4, no.1, pp. 101-107,2018.

[36] H.P. Hutter, *Comparison of classic and hybrid HMM approaches to speech recognition over telephone lines*, Zürich, 1997.

[37] InvenSens , MPU 6000 and MPU 6050 Product Specification. Available at: https://www.invensense.com/wpcontent/uploads/2015/02/MPU-6000-Datasheet1.pdf ,Accessed 09 Jun. 2018

[38] P. Jia, H. Huosheng, H. Hu, T. Lu, and K. Yuan. "Head Gesture Recognition for Hands-free Control of an Intelligent Wheelchair". *Industrial Robot: An International Journal*, vol.34, no.1, pp.60–68, 2007.

[39] S. Jia, J. Yan, J. Fan, X. Li, L. Gao, "Multimodal Intelligent Wheelchair Control Based on Fuzzy Algorithm", *Proceeding of the IEEE International Conference on Information and Automation,* Shenyang, China, June 2012

[40] S. Ju, Y. Shin, and Y. Kim. "Intelligent wheelchair (IW) interface using face and mouth recognition". *In Proceedings of the 13th international conference on Intelligent user interfaces*, pp.307–314, 2009.

[41] k. kannan, j. selvakumar, "arduino based voice controlled robot" , *international research journal of engineering and technology* ,vol.2, no.1 , pp.2395-0072, 2015.

[42] G. Krishnamurthy, M. Ghovanloo, "*Tongue drive*: A *tongue operated magnetic sensor based wireless assistive technology* for *people* with *severe disabilities". In: Proceedings of the* IEEE *International Symposium on Circuits and Systems (ISCAS)*, Island of Kos, Greece, pp. 5551–5554,2006.

[43] M. Krishnan, M. Mariappan," EEG-Based Brain-Machine Interface (BMI) for Controlling Mobile Robots: The Trend of Prior Studies", *IJCSEE*, vol. 3, no.2, pp.159-165, 2015.

[44] A. Lankenau and T. R¨ofer. "Mobile robot self-localization in large-scale environments". *In Proceedings of the IEEE International Conference on Robotics and Automation*, pp.1359–1364, 2002.

[45] S.P. Levine, D.A. Bell, L.A. Jaros, R.C. Simpson, Y. Koren, and J. Borenstein. "The navchair assistive wheelchair navigation system". *IEEE Transactions on Rehabilitation Engineering*, vol.7, pp.452–463, 1999.

[46] G. Li, Y. He, Y. Wei, S. Zhu, Y. Cao "The MEMS gyro stabilized platform design based on Kalman Filter*" International Conference on Optoelectronics and Microelectronics (ICOM)*, 2013

[47] V. R. Madarasz, L. C. Heiny, R. F. Cromp, and N.M. Mazur. "The design of an autonomous wheelchair for the disabled". *IEEE Robotics and Automation*, vol.2, no.3, pp.117–126, 1986.

[48] T. Mahalakshmi, P. Swaminathan and R. Muthaiah"An Overview of Template Matching Technique in Image Processing". *Research Journal of Applied Sciences, Engineering and Technology*, vol.4, no.24, pp. 5469-5473, 2012.

[49] J. Makhoul , R. Schwartz," State of the Art in Continuous Speech Recognition", *Proceedings of the National Academy of Sciences of the United States of America*, vol. 92, pp. 9956-9963, 1995.

[50] C. Mandel, K. Huebner, and T. Vierhuff. "Towards an autonomous wheelchair: Cognitive aspects in service robotics". *In Proceedings of Towards Autonomous Robotic Systems (TAROS'05)*, pp.165–172, 2005.

[51] C. Martens, N. Ruchel, B. Borgerding, O. Ivlev and A. Gräser. "Friend: Functional robot arm with user friendly interface for disabled people". *5th European Conference for the Advancement of Assistive Technology*, pp 57–65, 1999.

[52] Microsoft Windows Speech Recognition, available at https://www.Windowscentral.com/how-set-speech-recognition-windows-10, accessed 24 June 2018.

[53] M.A. Mikulski, "Electromyogram control algorithms f or the upper limb single-dof owered exoskeleton.", *IEEE: 4th International Conference on Human System Interactions (HSI)*, 2011.

[54] R. Morales, V. Feliu, A. Gonz´alez, and P. Pintado. "Kinematic model of a new staircase climbing wheelchair and its experimental validation". *International Journal of Robotic Research*, vol.25. no.9, pp.825–841, Sep 2006.

[55] J. S. Nguyen. "A *Smart Wheelchair System using a Combination of Stereoscopic and Spherical Vision Cameras"*. PhD thesis, University of Technology, Sydney, 2012.

[56] P.D. Nisbet, J. Craig, J.P. Odor, and S. Aitken. "smart" wheelchairs for mobility training". *Technology and Disability*, vol. 5, no. 1, pp. 49-62, 1996.

[57] M.D.Patel, S. Lakhapati, R. Chavan,and A. Pathak," Android Controlled Wheelchair", *International Journal for Research in Emerging Science and Technology*, vol. 4, no.1, 2017.

[58] N. Perveen, I. Bhardwaj, and D. Kumar. "An Overview on template Matching Methodologies and its Applications".*International Journal of Research in Computer and Communication Technology*, vol .2, no.10, 2013.

[59] G. Pires, U. Nunes, and A. T. de Almeida. "Robchair - a semi-autonomouse wheelchair for disabled people". *In Proceedings of the 3rd IFAC Symposium on Intelligent Autonomous Vehicles (IAV´98)*, 1998.

[60] E. Prassler, J. Scholz, and P. Fiorini. "A robotics wheelchair for crowded public environment". *Robotics Automation Magazine, IEEE*, vol.8.no.1,pp.38 –45,2001

[61] A. Pruski, M. Ennaji, and Y. Mor`ere. “Vahm : A user adapted intelligent wheelchair”. *In IEEE Conference on Control Application*, Scotland, Sep 2002.

[62] D. Purwanto, R. Mardiyanto and K. Arai, "Electric wheelchair control with gaze direction and eye blinking," *Artificial Life* and *Robotics,* vol.14, no.3, pp.397-400*,* 2009.

[63] R. S. Rao, K. Conn, S.H. Jung, J. Katupitiya, T. Kientz, V. Kumar, J. Ostrowski, S. Patel, and C. J. Taylor. “Human robot interaction: Application to smart wheelchairs”. *In International Conference on Robotics and Automation (ICRA’02)*, May 2002.

[64] John Sharp ,*Microsoft Visual C# Step by Step 9th Edition*, Kindle, 2018.

[65] M. Rowe, *Sensor basics: Types, functions and Applications*, EDN network, 2013

[66] V. Santhanam and V. Viswanathan, “Smartphone accelerometer controlled automated wheelchair”, *Proceedings of the 3rd International Conference on Electronics, Biomedical Engineering and its Applications*, Hong Kong, China, pp. 57 - 61, 2013.

[67] L. Sathishbalaji, R. Bhakkiyalakshmi,” Electric Wheelchair Controlled by EMG Signals with Obstacle Detection”, *IJEDR*, vol. 2, no. 2, pp. 1409-1412, 2014.

[68] L. Silva. “Head gestures recognition”. *In Proceedings of the International Conference on Image Processing*, vol.3, pp.266–269, 2002.

[69] D. Simon, “Kalman Filtering”, *Embedded Systems Programming*, vol. 14, no. 6, pp. 72-79, June 2001

[70] R C. Simpson and Simon P. Levine, “Voice Control of a Powered Wheelchair”, *IEEE transactions on Neural System and Rehabilitation Engineering,* vol.10, no.2, , pp. 122-125, June 2002.

[71] Store.arduino.cc , Arduino Mega 2560 Rev, Available at: https://store.arduino.cc/usa/ arduino-mega-2560-rev3 , Accessed 24 June 2018.

[72] Store.arduino.cc , Arduino UNO Rev3, Available at: https://store.arduino.cc/usa /arduino-uno-rev3 , Accessed 24 June 2018.

[73] F. A. Surahio, A. K. Jumani , S. Talpur, **“**An Approach to accept input in Text Editor through voice and its Analysis, designing, development and implementation using Speech Recognition”, *IJCSNS International Journal of Computer Science and Network Security*, vol.16, no.3, 2016 .

[74] S. D. Suryawanshi , J. S. Chitode , S. S. Pethakar,” Voice Operated Intelligent Wheelchair”, *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 3, no. 5, pp. 487-490, May 2013.

[75] D. Vanhooydonck, E. Demeester, A. Hintemann, J. Philips, G. Vanacker, H. Van Brussel, and M. Nuttin, “Adaptable navigational assistance for intelligent wheelchairs by means of an implicit personalized user model”, *Robotics and Autonomous Systems, ROBOT* , vol. 58, pp. 963–977, 2010.

[76] P. Viola, and M. Jones, “Robust Real-time Face Detection”, *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137-154, 2004.

[77] Wikipedia.org, available at: https://en.wikipedia.org/wiki/ List_of _speech_ recognition _ software , accessed 23 June 2018.

[78] World Health Organization, World report on disability, available at: http://www.who. int/disabilities/world_report/2011/report.pdf , accessed 18 June 2018.

[79] S.Yadav, P. Sheoran,“Smart Wheelchairs - A literature review”, *International Journal of Innovative and Emerging Research in Engineering,* vol. 3, no. 2, pp. 86-90, 2016.

[80] S. Yathunanthan, L.U.R. Chandrasena, A. Umakanthan, V. Vasuki, ”Controlling a wheelchair using EOG Signal”, *IEEE*, pp. 283-288, Dec. 2015

[81] S. Yokota,Y. Ohyama,H. Hashimoto, J. HuaSHE, H. Kobayashi, and P. Blazevic,” Development of Electric Wheelchair Controlled by Human Body Motion”, *SICE Annual Conference*, 2008, Japan.

[82] A.Youssef, C. Guillaume, and R. Nicolas, “Omnidirectional photometric visual path following for wheelchair autonomous driving”, *1st Healthcare Technology Days, CARETECH*, Dec 2016, Rouen, France.

[83] Y Zhang, J Zhang, Y Luo, “A novel intelligent wheelchair control system based on hand gesture recognition”, *IEEE/ICME International Conference on Complex Medical Engineering*, pp.334-339, China,2011.