

# **DEDICATION**

We dedicate to everyone who believes in the idea of this thesis.

Dedicate to everyone helped us to compile this thesis.

To our supervisor who spent a remarkable Time to help and put us in the right track.

Finally, we do appreciate every help we received from everyone.

# ACKNOWLEDGMENT

First and foremost, we would like to thank the Supervisor Dr. Mohammed Yagoub for his support, outstanding guidance and encouragement throughout our senior project.

We would like to thank our family, especially our parents, for their encouragement, patience, and assistance over the years. We are forever indebted to our parents, who have always kept us in their prayers.

# ABSTRACT

Sign language is a primary means of communication amongst the deaf peoples, most normal peoples do not understand sign language, and this creates a big gap in communication between them. There are various studies on sign language recognition (SLR) to solving this problem, most of them use accessories like colored gloves and accelerometers for data acquisition or used system that required complex environmental setup to operate. The Proposed work to recognize the Arabic sign language was design an integrated system to translate this Arabic sign language to Arabic text by using the Kinect sensor. In this thesis, the Kinect sensor is used to acquire the gestures from the signer in a real-time video format. Kinect provides 3D depth information that used to show points and sticks in the skeleton that related to the main joints of the signer by using Randomization Decision Forest and Mean Shift algorithms. After that, the movement of the skeleton joints is tracked. Then, Features from skeleton frame extracted by using calculations and Scale-Invariant Feature Transform algorithm, this features used to match with the predefined gestures set. If the current skeleton frame matches the predefined gesture pattern, the corresponding word for the gesture is shown as text, this called classification stage and it performed by Dynamic Time Warping algorithm.

The system is able to recognize 6 Arabic words that make up the dataset and configure sentences that related to previous words with acceptable accuracy and precision.

## المستخلص

لغة الإشارة هي وسيلة التواصل الأساسية بين الصم والبكم ، معظم الأشخاص يواجهون صعوبة في فهم لغة الإشارة مما يخلق فجوة كبيرة في التواصل بينهم. هناك العديد من الدراسات حول ترجمة لغة الإشارة لحل هذه المشكلة، معظم هذه البحوث تتمحور حول استخدام الملحقات مثل القفازات الملونة وحساس مقياس التسارع للحصول على البيانات أو تتمحور حول الانظمة التي تتطلب الإعداد المعقد لبيئة العمل. النموذج المقترح لترجمة لغة الإشارة العربية هو تصميم نظام متكامل لترجمة هذه اللغة إلى نص عربي باستخدام جهاز استشعار (كاميرا) كينكت. في هذه الأطروحة تم استخدام جهاز استشعار كينكت ليقوم بالنقاط الحركات التي يقوم بها المستخدم على شكل فيديو في الوقت الحقيقي. توفر كينكت تقنية 3D depth التي تمكن من التقاط العمق و تستخدم هذه المعلومات لإظهار الهيكل العظمي والذي توضح فيه المفاصل الرئيسية لجسم المستخدم وذلك باستخدام خوارزمية Mean Shift وخوارزمية Randomization Decision Forest . بعد ذلك يتم تتبع حركة مفاصل الهيكل العظمي ثم يتم استخراج المميزات الأساسية لكل حركة باستخدام الحسابات وخوارزمية Scale-Invariant Feature Transform ، ثم يتم مقارنة هذه المميزات المستخرجة مع مجموعة مميزات معرفة مسبقاً في قاعدة بيانات ، اذا حدث تطابق بين الميزتين يتم عرض الكلمة المقابلة للحركة كنص عربي مكتوب ، وهذا ما يسمى مرحلة التصنيف ويتم ادائه بخوارزمية Dynamic Time Warping.

تم تجريب هذا النظام للتعرف على ست كلمات شائعة في لغة الإشارة العربية وتم الحصول على نتائج مرضية.

# TABLE OF CONTENTS

Dedication.....	I
Acknowledgment.....	II
Abstract.....	III
المستخلص.....	IV
Table of Contents.....	V
List of Tables.....	VIII
List of Figures.....	IX
List of Abbreviation.....	XI

## CHAPTER ONE

### INTRODUCTION

1.1 Introduction.....	1
1.2 Problem Statement.....	4
1.3 Objectives.....	4
1.4 Methodology.....	4
1.5 Thesis Layout.....	6

## CHAPTER TOW

### LITERATURE REVIEW

2.1 Sensor Based Data.....	7
2.2 Vision Based Computer.....	7

## CHAPTER THREE

### THEORETICAL BACKGROUND

3.1 Skeleton Capturing Device (Microsoft Kinect).....	11
3.1.1 Technological Overview.....	11
3.2 Infer Body Position by Kinect Camera.....	13
3.2.1 Randomized Decision Forest algorithm.....	13
3.2.2 Mean Shift Algorithm.....	15
3.3 Dynamic Time Warping (DTW) Algorithm.....	16

## **CHAPTER FOUR METHODOLOGY**

4.1 Project Aim .....	18
4.2 Initial Concept of Research Method .....	18
4.2.1 Data Glove .....	18
4.2.2 Software .....	19
4.2.2.1 MATLAB with Web Camera.....	19
4.2.2.2 Visual Studio with Web Camera.....	19
4.2.2.3 Visual Studio with Kinect Camera.....	20
4.3 The Proposed System.....	22
4.3.1 Preparing the Programming Environment.....	23
4.3.2 Process of ArSL Recognition System .....	23
4.3.2.1 Capture the signer’s video.....	23
4.3.2.2 Obtain the Depth Image .....	24
4.3.2.3 Skeletonize Data.....	25
4.3.2.4 Skeleton Tracking .....	27
4.3.2.5 Feature Extraction .....	28
4.3.3 Building Database and Classification .....	29
4.3.3.1 Classification.....	29
4.3.4 User Interface Design.....	31

## **CHAPTER FIVE RESULTS AND DISCUSSION**

5.1 Dataset.....	32
5.2 The Environment Factors .....	32
5.3 Depth Image and Skeletal Tracking.....	32
5.4 Experiments Results.....	34
5.4.1 Build Up The Sentences.....	37
5.5 Number of Attempts.....	42
5.6 Time of Response.....	44

**CHAPTER SIX**  
**CONCLUSION AND FUTURE WORK**

6.1 Summary and Advantages..... 49  
6.2 Limitations and Future Work..... 50

**References**

**Appendix**

# LIST OF TABLES

Table 4.1 illustrates the sentences used in this system.....	31
Table 5.1 skeleton frame of different signs.....	40
Table 5.2 the maximum number of attempts which system responds to the sign.....	43
Table 5.3 the time of response of the system for each sign.....	44



# LIST OF FIGURES

Figure 1.1 block diagram of recognition system.....	5
Figure 3.1 Component of the Kinect Sensor.....	12
Figure 3.2 Depth sensor physical limit.....	12
Figure 3.3 Example of classification tree.....	14
Figure 3.4 Kinect uses Randomize decision Forest.....	14
Figure 3.5 Dynamic time warping.....	16
Figure 3.6 local constraints.....	17
Figure 4.1 Region of Interest.....	20
Figure 4.2 block diagram of initial recognition ArSL to Arabic letters system.....	21
Figure 4.3 Overall Segmentation Process design.....	22
Figure 4.4 four basic stages of proposed system.....	23
Figure 4.5 kinect sensor hardware.....	24
Figure 4.6 obtain the depth information of the object by the kinect sensor.....	25
Figure 4.7 the skeletonized data phase.....	26
Figure 4.8 XYZ plane that describe joint position.....	26
Figure 4.9 positions of 20 joints in the skeleton.....	27
Figure 4.10 describe matching real-time video with features in dataset.....	30
Figure 4.11 gesture recognition of "هل انت جائع؟".....	31
Figure 5.1 Depth image.....	33
Figure 5.2 skeletal detection and tracking.....	33
Figure 5.3 gesture recognition of "مرحبا".....	34
Figure 5.4 gesture recognition of "ماذا".....	35
Figure 5.5 gesture recognition of "أنت".....	35
Figure 5.6 gesture recognition of "عُمر".....	36
Figure 5.7 gesture recognition of "جائع".....	36
Figure 5.8 gesture recognition of "اسم".....	37
Figure 5.9 gesture recognition of "كم عمرك؟".....	38
Figure 5.10 gesture recognition of "ما اسمك؟".....	38

Figure 5.11 gesture recognition of "هل انت جائع؟"	39
Figure 5.12 shown the Kinect sensor is just response to the close person.	39
Figure 5.13 the number of attempts which system responds to each sign	43
Figure 5.14 time response of "مرحبا" sign	45
Figure 5.15 time response of "ماذا" sign	45
Figure 5.16 time response of "أنت" sign	46
Figure 5.17 time response of "عُمر" sign	46
Figure 5.18 time response of "جائع" sign	47
Figure 5.19 time response of "اسم" sign	47

# LIST OF ABBREVIATIONS

ADC	Analog to Digital Converter
AI	Artificial Intelligence
ArSL	Arabic Sign Language
BSD	Berkeley Software Distribution
Db	Decibels
DCT	Discrete Cosine Transformation
DTW	Dynamic Time Warping
HCI	Human Computer Interaction
HGR	Hand Gesture Recognition
HMM	Hidden Markov Model
HOG	Histogram Of Oriented Gradients
IR	Infrared
MLL	Machine Learning Library
MLP	Multi Layer Perceptron
NUI	The Natural User Interface

OpenCV	Open Source Computer Vision Library
PCA	Principle Component Analysis
RGB	Red, Green, and Blue
ROI	Region of Interest
RPCA	Recursive Principle Components Analysis
SDK	Software Development Kit
SLR	Sign Language Recognition
WPF	Windows Presentation Foundation

# CHAPTER ONE

## INTRODUCTION

### 1.1 Introduction

Human-Computer Interaction (HCI) is getting increasingly important as computer's influence on our lives and becoming more and more significant. With the advancement in the world of computers, the already-existing HCI devices (the mouse and the keyboard for example) are not satisfying the increasing demands anymore. Designers are trying to make HCI faster, easier, and more natural. To achieve this, Human-to-Human Interaction techniques are being introduced into the field of HCI. One of the most fertile Human-to-Human Interaction fields is the use of hand gestures. People use hand gestures mainly to communicate and to express ideas.

Sign Language Recognition (SLR) is an important part of the larger application field of Hand Gesture Recognition (HGR) in HCI. Many researchers concentrated on translating from and to SLR due to its impact on the society. Different approaches have been customized by different researchers for the recognition of various sign language hand gestures. Some of the approaches were vision or data glove based and soft computing approaches like Artificial Neural Network, Fuzzy logic, Genetic Algorithm, Principle Component Analysis (PCA), Canonical Analysis, etc... [1]

The importance of using hand gestures for communication becomes clearer when sign language is considered. The Sign Language is defined as "the manual representation of language relying on the use of signed vocabulary to represent concepts"[2]. Sign language as a kind of gestures is one of the most natural ways of communication for most people in the deaf community.

Over 5% of the world's population – 360 million people – has disabling hearing loss (328 million adults and 32 million children). Disabling hearing loss refers to hearing loss greater than 40 decibels (dB) in the better hearing ear in adults and a hearing loss greater than 30 dB in the better hearing ear in children [3]. Recently, there has been a

serious need for the deaf community in the Arab world to be able to communicate and integrate with the rest of the society. The deaf community has been accustomed to conducting most of its daily affairs in isolation and only with people capable of understanding sign language. This isolation deprives this sizable segment of the society from proper socialization, education, and aspiration to career growth. This lack of communications hinders the deaf community from deploying their talents and skills in benefiting the society at large. For help deaf people to communicate with non-deaf people in the Arab world, it should build a system that does the function of translation this Arabic Sign Language (ArSL) to Arabic text easy to read and understand by using Kinect camera and machine learning techniques.

The earlier work in this field involved the use of complex gloves in this method data is collected by one or more data gloves, gloves must be worn and wear some device with a load of cables connected to the computer, data gloves are not practical, costly and need some sort of setup. Then after those many researchers have used a digital webcam since it is low cost and needs almost no setup. Working with a webcam in sign language recognition systems may suffer from some problems. One of them is that complex background adds false skin-like regions. Also, it would be very difficult if another person appears in the scene and it would fail with changing a background, especially when using background subtraction technique [4]. Finally, some researchers have used an improved method in sign language recognition that tries to translate a continuous stream of words by using the Kinect camera.

Kinect Camera is the device ability to track human joints through motion capture system has been tested, the experiment proved that Kinect system is able to recognize the various joints of the body in the most controlled poses. It is becoming increasingly popular in many areas aside from entertainment, including human activity monitoring and rehabilitation [5].

The core of the Kinect API is the Natural User Interface (NUI). Through it, a developer can access to Audio data stream out by the audio stream, and Color image data and depth image data streamed out by the color and depth streams and applied it in his application. In addition to the hardware capabilities, the Kinect software runtime

implements a software pipeline that can recognize and track a human body, and the runtime converts depth information into the skeleton joints in the human body [6].

Kinect has not been employed in ArSLR; therefore, a new dataset composed of ArSL words captured using the Kinect camera was developed. So we will use the computer vision and machine learning to build a database.

The Computer vision is a field that includes methods for acquiring, processing, analyzing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information and the Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to explicitly program. There are many security applications of machine learning, e.g. for access control, use face recognition as one of its components. That is, given the photo (or video recording) of a person, recognize who this person is. In other words, the system needs to classify the faces into one of many categories (Alice, Bob, Charlie...) or decide that it is an unknown face [7]. Similar, it can be used for sign recognition.

OpenCV (Open Source Computer Vision Library) is an open-source BSD-licensed library that includes several hundreds of computer vision algorithms that can be used with different languages C, C++, Java, Python, etc. some of the OpenCV features are Camera calibration (finding and tracking calibration patterns), Motion analysis (optical flow, motion segmentation, tracking) and Object recognition (Eigen-methods, Hidden Markov Model (HMM)). The OpenCV is playing important role in the growth of computer vision by enabling thousands of people to do more productive work in vision. Focusing on real-time visibility and helps students and professionals execute projects efficiently.

One of OpenCV's goals is to provide a simple-to-use computer vision infrastructure that helps people build fairly sophisticated vision applications quickly. The OpenCV library contains over 500 functions that span many areas in vision, including factory product inspection, medical imaging, security, user interface, camera calibration, stereo vision, and robotics. Because computer vision and machine learning often go hand-in-hand, OpenCV also contains a full, general-purpose Machine Learning Library (MLL). This sub-library is focused on statistical pattern recognition and clustering. The MLL is

highly useful for the vision tasks that are at the core of OpenCV's mission, but it is general enough to be used for any machine learning problem [8].

## **1.2 Problem Statement**

Deaf people using sign language as native because it is the only way for them to clarify their needs. It is an essential part of their life. Being unable to understand deaf people makes a big gap between them and non-deaf people and the translation of (ArSL) in Sudan or Arab world is usually conventional by using person capable of understanding sign language, this may cause embarrassment to the deaf in most situations. In addition to the fact that most of the time the interpreter may not be available.

The above fact, lead to developing the conventional technique to be more easy and available in anywhere and developing an automatic sign language recognition system, which can accomplish the need of hearing impaired people.

## **1.3 Objectives**

The objective of this project is to build an integrated system to Recognize signs from Arabic Sign Language to Arabic text using Kinect camera and computer vision techniques. The proposed system will translate the video of sign to text. The motive of this work is to provide a real-time interface so that signers can be able to easily and quickly communicate with non-signers.

The specific objectives are to learn how to capture video by Kinect camera, skeletal tracking, feature extraction, and classification.

## **1.4 Methodology**

The proposed system is composed of several stages shown in figure (1.1). These stages are video capturing, obtained depth image, feature extraction, image classification and finally recognition the signs into Arabic text.



The recognition system begins with capturing of active signer who is the nearest person to Kinect sensor. The Kinect sensor used to acquire the gestures from the signer in a real-time video format. Kinect provide a depth data that used to show where the signer's Limbs are and show them as points and sticks. The next stage is tracking the movement of the skeleton joints. Then feature extraction by using some algorithms which are used as a descriptor not only for human body detection but also for any shape. After feature-extraction, it is time to classification these features and recognition the signs into Arabic text. All of these processes are done by C# code.

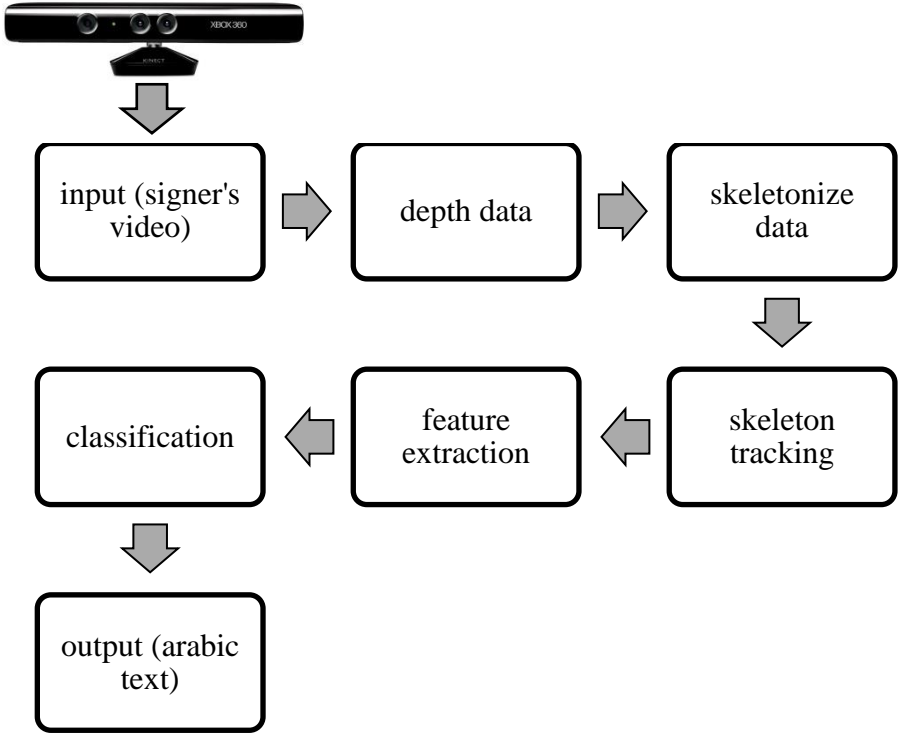


Figure 1.1: block diagram of recognition system

## **1.5 Thesis Layout**

This project is organized of 5 chapters there are: Chapter one include the introduction of the thesis, Chapter 2 provides review about the previous studies for the main problem of translation sign language, Chapter 3 gives a brief description of the proposed methodology of the recognition system, Chapter 4 provide experiment result and discussion this result, Chapter 5 Conclusion Contains comments about the project with focus on its State of completion with regards to its aim and objectives.

# CHAPTER TWO

## LITERATURE REVIEW

Communication is an important thing in life, which can be done many ways to communicate, such as by speaking through oral or sign language.

Sign language is basic alternative communication method between deaf people. There are several dictionaries of word or signal letters have been defined to make this communication possible. Sign language used by the deaf and speech impaired is difficult to understand by the general public they feel ostracized by the surrounding environment. Progress in the field of pattern recognition systems and Human Computer Interaction (HCI) promises to the automatic sign language interpreter. Research has been conducted in order to generate tools to help to translate sign language into text.

Researchers in the field of sign language are categorized into two technologies vision based computer (computer vision) and based on the sensor data.

### **2.1 Sensor Based Data**

Researchers Priyanka Lokhande, Riya Prajapati and Sandeep Pansare are building a system that consists of a glove that will be worn by a deaf person to facilitate the communication with the normal person. It translates the hand gestures to corresponding words using flex sensors and 3-axis accelerometer. The signals are converted to digital data using comparator circuits and Analog to Digital Converter (ADC) of microcontroller ARM LPC 2138. The microcontroller matches the binary combinations with the data of the databases and produces the speech signal. The output of the system is displayed using the speaker and LCD [9].

### **2.2 Vision Based Computer**

Researchers M.AL-Rousan, K.Assaleh, A.Tala'a are writing paper that introduces the first automatic Arabic sign language (ArSL) recognition system based on hidden Markov models (HMMs). A large set of samples has been used to recognize 30 isolated

words from the Standard Arabic sign language. The system operates in different modes including offline, online, signer-dependent, and signer-independent modes. Experimental results on using real ArSL data collected from deaf people demonstrate that the proposed system has high recognition rate for all modes. For signer-dependent case, the system obtains a word recognition rate of 98.13%, 96.74%, and 93.8%, on the training data in offline mode, on the test data in offline mode, and on the test data in online mode respectively. On the other hand, for signer-independent case the system obtains a word recognition rate of 94.2% and 90.6% for offline and online modes respectively. The system does not rely on the use of data gloves or other means as input devices, and it allows the deaf signers to perform gestures freely and naturally [10].

Researchers M.F.Tolba, Ahmed Samir, Magdy Aboul-Elaare focusing on how to recognize the real-time connected sequence of gestures using the graph-matching technique, also how the continuous input gestures are segmented and classified. Graphs are a general and powerful data structure useful for the representation of various objects and concepts. This work is a component of a real-time Arabic Sign Language Recognition system that applied pulse-coupled neural network for static posture recognition in its first phase. This work can be adapted and applied to different sign languages and other recognition problems [11].

Researchers T. Shanableh and K. Assaleh, M. Fanaswala, F. Amin, H. Bajaj are presenting a solution for user-independent recognition of isolated Arabic sign language gestures. The video-based gestures are preprocessed to segment out the hands of the signer based on color segmentation of the colored gloves. The prediction errors of consecutive segmented images are then accumulated into two images according to the directionality of the motion. Different accumulation weights are employed to further help preserve the directionality of the projected motion. Normally, a gesture is represented by hand movements; however, additional user-dependent head and body movements might be present. In the user-independent mode, they seek to filter out such user-dependent information. This is realized by encapsulating the movements of the segmented hands in a bounding box.

The encapsulated images of the projected motion are then transformed into the frequency domain using Discrete Cosine Transformation (DCT). Feature vectors are formed by applying Zonal coding to the DCT coefficients with varying cutoff values. Classification techniques such as nearest neighbors (KNN) and polynomial classifiers are used to assess the validity of the proposed user-independent feature extraction schemes. An average classification rate of 87% is reported [12].

A.S.Elons, Menna Ahmed and Hwaidaa Shedid presents a study on an ArSL database is performed to conclude that the 6 main facial expressions are essential to recognize the sign.

A developed system used to classify these expressions accomplished 92% recognition rate on 5 different people. The system employed already existing technical methods such as Recursive Principle Components Analysis (RPCA) for feature extraction and Multi-layer Perceptron (MLP) for classification. The main contribution of this paper is employing the developed module and integrating it with an already existing hand sign recognition system. The proposed system enhanced the hand sign recognition system and raised the recognition rate from 88% to 98% [13].

Noha A. Sarhan, Yasser EI-Sonbaty, Sherine M. Youssef are designing a proposed system that combines skeletal data and depth information for hand tracking and segmentation, without relying on any color markers, or skin color detection algorithms. The extracted features describe the four elements of the hand that are used to describe the phonological structure of ArSL: articulation point, hand orientation, hand shape, and hand movement. Hidden Markov Model (HMM) was used for classification using ten-fold cross-validation, achieving an accuracy of 80.47%. Singer independent experiments resulted in an average recognition accuracy of 64.61% [14].

Ayman Hamed, Nahla A. Belal , Khaled M. Mahar published paper about Arabic sign language alphabet recognition based on HOG-PCA using Microsoft Kinect in complex backgrounds. They use a Kinect camera to capture signer's video and then segmentation the only signer's right hand by many processes. After that, features extracted from the right hand are used to train a support vector machine classifier. The system is able to recognize the 30 Arabic alphabets with an accuracy of 99.2% [4].

The Proposed work of this thesis was design an integrated system to translate signs from Arabic Sign Language to Arabic text by using Kinect sensor because it is lower in cost than other 3D depth sensors and due to its depth information that provides, it can keep track of object distance from the camera allowing for a good recognition process. When Human gestures enter to the system, the frame of the skeleton data of the user is tracked with the joints by a kinect sensor. Then the skeleton frame matches with the predefined gesture pattern and the corresponding word for the gesture will appear.

# **CHAPTER THREE**

## **THEORETICAL BACKGROUND**

In this chapter, specifications and usage of the Kinect sensor which is the input device of this system, an overview of Mean Shift, Randomize Decision Tree and Dynamic Time Warping (DTW) algorithms which are used in the scope of this study are presented.

### **3.1 Skeleton Capturing Device (Microsoft Kinect)**

Kinect is an input device developed by Microsoft which aims to sense motion and voice. It is used as the input device of the proposed system. In this section, technical specifications and usage of it are given.

#### **3.1.1 Technological Overview**

The Kinect camera as hardware is consisting of three main components they are the RGB camera, an infrared (IR) emitter, and a multi-array microphone (figure 3.1).

The first component is RGB camera that stores three channel data in a 1280x960 resolution. This makes capturing a color image possible. An infrared (IR) emitter and an IR depth sensor are the second component of kinect camera. The emitter emits infrared light beams and the depth sensor reads the IR beams reflected back to the sensor. The reflected beams are converted into depth information measuring the distance between an object and the sensor. This makes capturing a depth image possible.

The last component is a multi-array microphone, which contains four microphones for capturing sound. Because there are four microphones, it is possible to record audio as well as find the location of the sound source and the direction of the audio wave.

A 3-axis accelerometer configured for a 2G range, where G is the acceleration due to gravity. It is possible to use the accelerometer to determine the current orientation of the Kinect [15].

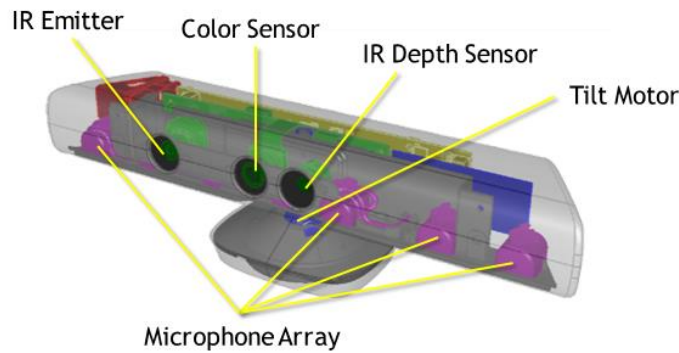


Figure 3.1: components of the kinect sensor

The depth camera can save 30 image frames with 640 x 480, 320 x 240 and 80 x 60 resolutions per second. Because data gathered by this sensor is used for depth-sensing it is also called as depth sensor. Angle of the view of the depth sensor is limited to 57.5 degrees horizontally and 43.5 degrees vertically. The range of the depth the sensor can measure is from 0.8 to 4 meters in the default mode and 0.4 to 3 meters in the near mode. Sweet spot, optimal interaction distance range with the depth sensor, is from 1.2 to 3.5 meters (see Figure 3.2 for visualizations) [6].

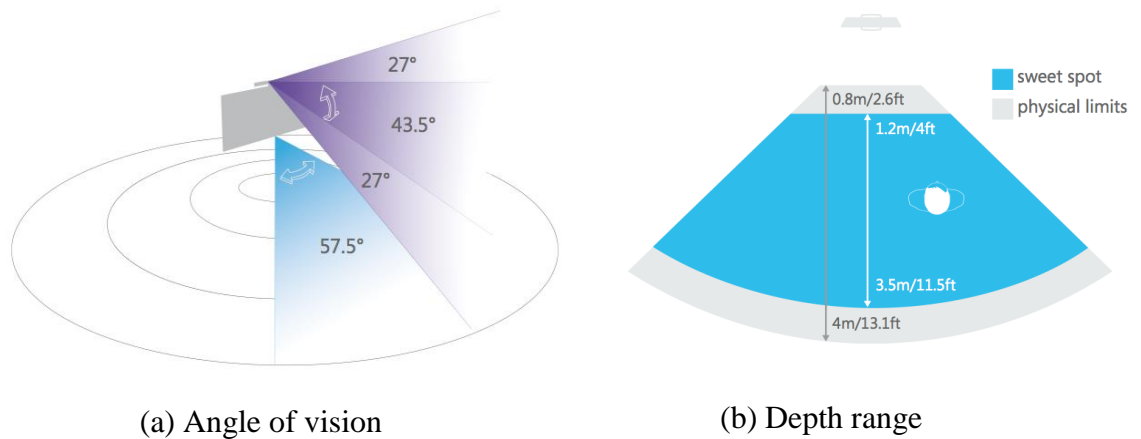


Figure 3.2: Depth sensor physical limits. (a) Describe the Angle of the view of the depth sensor. (b) Describe the range of the depth the sensor.



Kinect is a device that Microsoft developed for the purpose of gaming. Microsoft has given the opportunity for programmers to develop different systems. Therefore, by using the features that Kinect SDK provided, it is possible to make new methods for the process instead of gaming only.

## 3.2 Infer Body Position by Kinect Camera

Body parts are inferred using a Randomized Decision Forest algorithm, learned from over 1 million training examples. It starts with 100,000 depth images with known skeletons (from a motion capture system). For each real image, render dozens more using computer graphics techniques. Then learn a Randomized Decision Forest, mapping depth images to body parts. After inferring body part, Kinect using Mean Shift algorithm to transform the body part image into a skeleton.

### 3.2.1 Randomized Decision Forest algorithm

A random forest is a classifier consisting of a collection of tree structured classifiers  $\{h(\mathbf{x}, \Theta_k), k=1, \dots\}$  where the  $\{\Theta_k\}$  are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input  $\mathbf{x}$  [16].

Given an ensemble of classifiers  $h_1(\mathbf{x}), h_2(\mathbf{x}), \dots, h_K(\mathbf{x})$ , and with the training set drawn at random from the distribution of the random vector  $Y, \mathbf{X}$ , define the margin function as

$$mg(\mathbf{X}, Y) = \text{avg } I(h_k(\mathbf{X})=Y) - \max_{j \neq Y} \text{avg } I(h_k(\mathbf{X})=j) \quad (3.1)$$

Where  $I(\bullet)$  is the indicator function. The margin measures the extent to which the average number of votes at  $\mathbf{X}, Y$  for the right class exceeds the average vote for any other class. The larger the margin, the more confidence in the classification. The generalization error is given by

$$PE^* = P_{\mathbf{X}, Y} (mg(\mathbf{X}, Y) < 0) \quad (3.2)$$

Where the subscripts  $\mathbf{X}, Y$  indicate that the probability is over the  $\mathbf{X}, Y$  space.

In random forests

$$hk(\mathbf{X}) = h(\mathbf{X}, \Theta k) \quad (3.3)$$

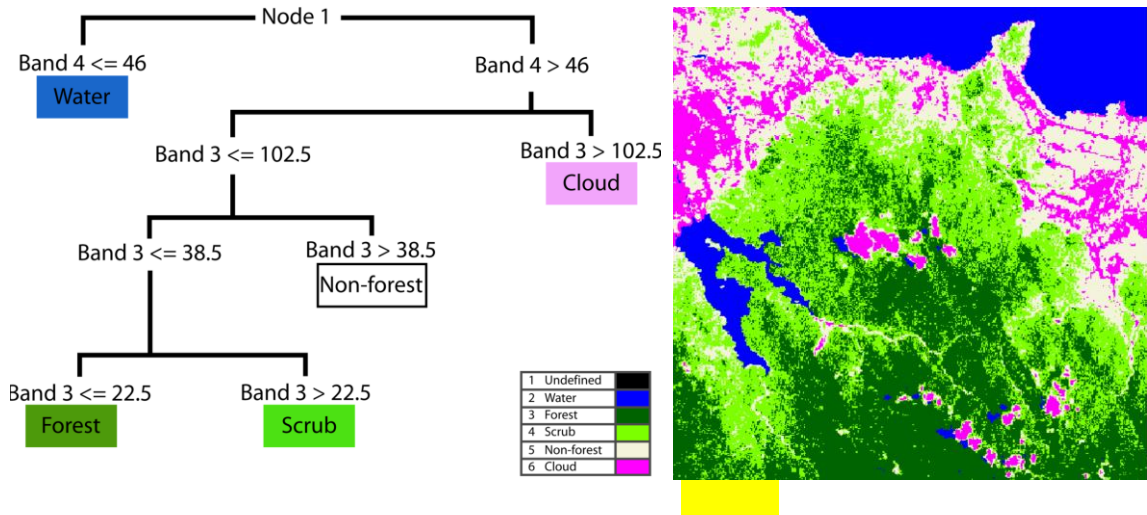


Figure 3.3: example of classification tree

A randomized decision forest is a more sophisticated version of the classic decision tree. Kinect actually uses a randomized decision forest as Randomized, that have Too many possible questions, so use a random selection of 2000 questions each time. And Forest, to learn multiple trees, to classify, add outputs of the trees, and outputs are actually probability distributions, not single decisions.

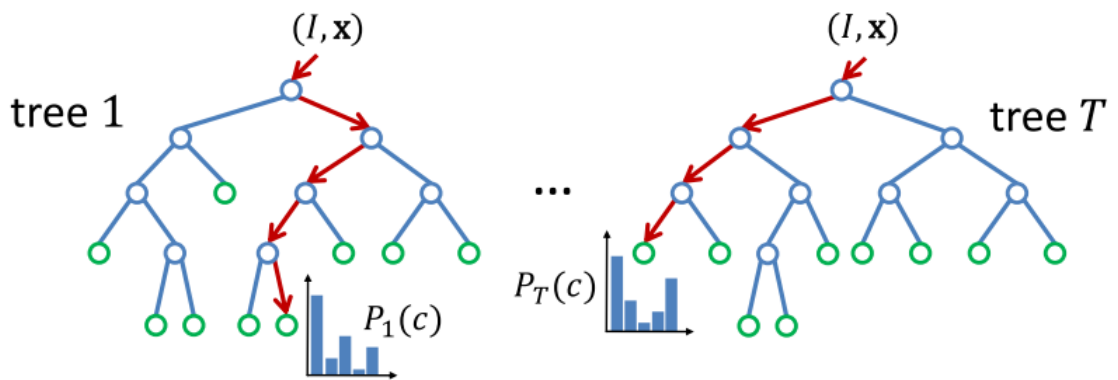


Figure 3.4: Kinect uses a randomized decision forest.

The Kinect used this algorithm and trained it with nearly million depth images with known skeletons. So, the classification of the skeleton is possible.

Learning the Kinect decision forest requires 24,000 CPU-hours, but takes only a day using hundreds of computers simultaneously

“To keep the training times down we employ a distributed implementation. Training 3 trees to depth 20 from 1 million images takes about a day on a 1000 core cluster.”[17]

### 3.2.2 Mean Shift Algorithm

Mean Shift algorithm is a nonparametric, iterative procedure that shifts each data to local maximum of density function.

It start with an initial estimate  $x$ . Let a kernel function  $K(x_i - x)$  be given. This function determines the weight of nearby points for re-estimation of the mean. Typically a Gaussian kernel on the distance to the current estimate is used,

$$K(x_i - x) = e^{-c|x_i-x|^2} \quad (3.4)$$

The weighted mean of the density in the window determined by  $K$  is

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x_i - x)x_i}{\sum_{x_i \in N(x)} K(x_i - x)} \quad (3.5)$$

Where  $N(x)$  is the neighborhood of  $x$ , a set of points for which  $K(x) \neq 0$

The difference  $(m(x) - x)$  is called mean shift in Fukunaga and Hostetler.[18]

The mean-shift algorithm now sets  $x \leftarrow m(x)$ , and repeats the estimation until  $m(x)$  converges.

Mean shift algorithm is recently widely used in tracking clustering. The Xbox team wrote a tracking algorithm that rejects “bad” skeletons and accepts “good” ones.

### 3.3 Dynamic Time Warping (DTW) Algorithm

Dynamic time warping (DTW) is an algorithm to find optimal alignment between two time series. It is used in biology, finance, medicine (ECG), speech technology, and query by humming.

If we have two time series Q and C:

$$Q = q_1, q_2, \dots, q_n$$

$$C = c_1, c_2, \dots, c_m$$

Construct  $(n \times m)$  matrix D with distances  $D_{ij} = d(q_i, c_j)$ .

Warping path W is a contiguous set of matrix elements

$$w_k = (i, j)_k \tag{3.6}$$

Define warping between Q and C

$$W = w_1, w_2, \dots, w_K$$

Where  $\max(n, m) \leq K \leq m + n - 1$

Find:

$$DTW(Q, C) = \min \sqrt{\sum w_k} \tag{3.7}$$

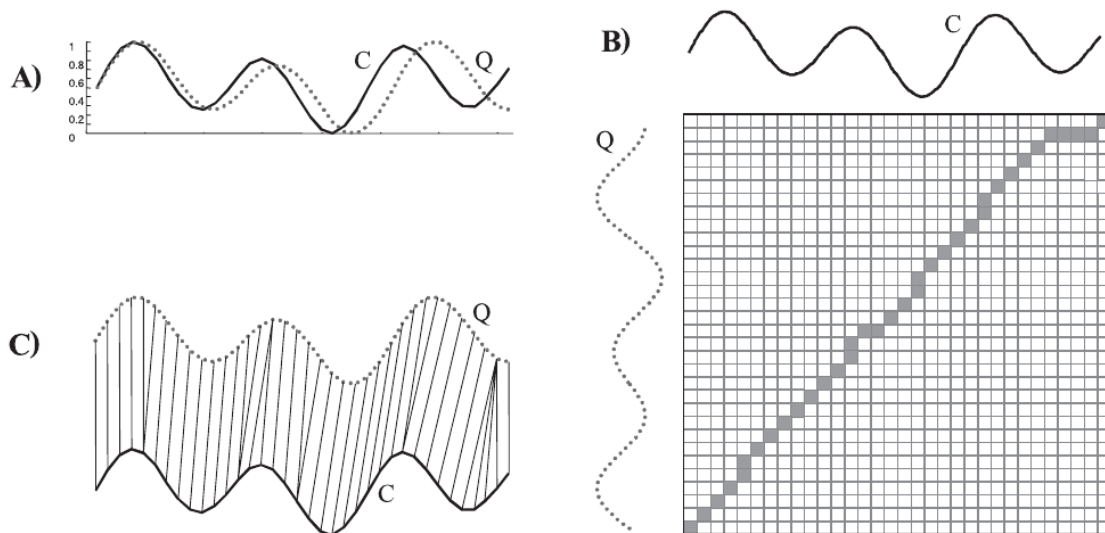


Figure 3.5: A) Two sequences Q and C that are similar but out of phase. B) To align the sequences, we construct a warping matrix and search for the optimal warping path, shown with solid squares. C) The resulting alignment.

There are some constraints on path

- boundary condition:

$$w_1 = (1, 1), \quad w_K = (n, m)$$

- Continuity:

$$\text{If } w_k = (i, j) \text{ and } w_{k-1} = (i', j') \text{ then } i - i' \leq 1 \text{ and } j - j' \leq 1$$

- Monotonicity

$$\text{If } w_k = (i, j) \text{ and } w_{k-1} = (i', j') \text{ then } i - i' \geq 0 \text{ and } j - j' \geq 0$$

Find optimal path

$$DTW(Q, C) = \min \sqrt{\sum w_k} \quad (3.7)$$

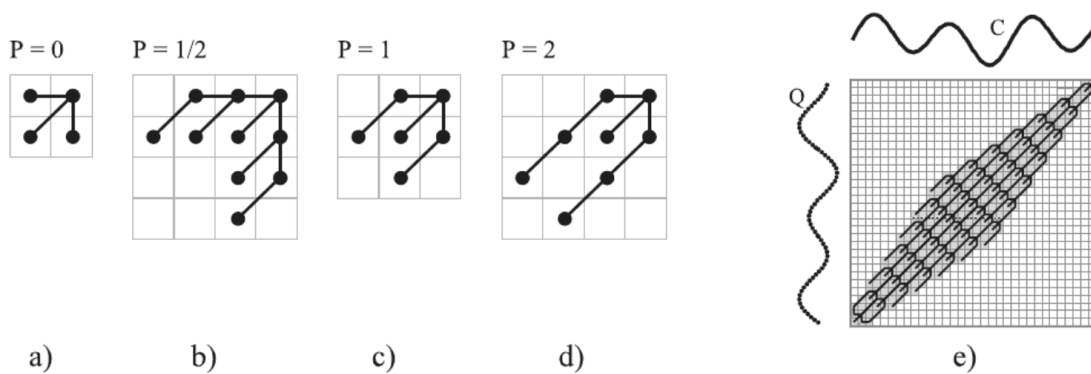


Figure 3.6: local constraints. a) No constraint, equivalent to  $\gamma(i, j) = d(i, j) + \min\{\gamma(i - 1, j - 1), \gamma(i - 1, j), \gamma(i, j - 1)\}$ . c)  $\gamma(i, j) = d(i, j) + \min\{\gamma(i - 1, j - 1), \gamma(i - 1, j - 2), \gamma(i - 2, j - 1)\}$ . d) local constraints to global gives e)

DTW was used to determine the similarity between the gestures done by the signer in real time, and that stored in data set. The trajectory and speed of individual joints involved in each sign are used as the feature vector and compared separately using DTW. The joint position information was obtained from the Kinect skeleton frames.

# CHAPTER FOUR

## METHODOLOGY

This chapter clarifies the methods and techniques that used in this project and the associated tools used during work in this project. The project aim is redefined and multiple goals derived in section 3.1. The project undergoes several stages; these stages include preparing Kinect sensor, skeleton tracking, feature extraction and classification.

### 4.1 Project Aim

The aim of the project is to design a software program that can translate Arabic sign language into Arabic text by taking the benefits of computer vision techniques. This goal is achieved by implementation the following:

- Preparation the programming environment to become compatible to open Kinect sensor.
- Write code that involves the whole process that should be done to getting the proper translation of hand gestures to Arabic letters and words.
- Building database.

### 4.2 Initial Concept of Research Method

The basic idea of the project is to translate Arabic sign language into Arabic words, so we had different of ideas to achieve this goal some of them were excluded. These primary ideas are explained in following:

#### 4.2.1 Data Glove

The first idea in this project is to design the Data glove. Data glove is a hardware glove that signer wear it. It consists of several sensors such as flex sensors that convert the hand gestures into Arabic letters. After searching and investigation found this idea is unpractical, costly and cannot be achieving the required efficiency. For these facts, the idea was changed and oriented into software field.

## **4.2.2 Software**

The entire world was oriented to the software in most fields especially in sign language field. Many researchers concentrated on translating from and to SLR like Chinese sign language recognition, American Sign Language recognition, and British sign language recognition, but in Arabic sign language recognition there were researchers published different papers but not applied in real life.

### **4.2.2.1 MATLAB with Web Camera**

In the software field, we had chosen MATLAB because of it easier in programming but we found some problems that prevent the project to perform as required. Those problems summarized in MATLAB is not open source, its license is costly, as well the MATLAB code is run slow especially in real time video processing. Due all of that has been changed from MATLAB into openCV.

OpenCV (Open Source Computer Vision Library) is released under a Berkeley software distribution (BSD) license and hence it's free for both academic and commercial use. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. It has more functions for computer vision than MATLAB. We prefer OpenCV rather than MATLAB because it has many features like the speed of execution; programs written in OpenCV run much faster than similar programs written in MATLAB. Further, the OpenCV is free.

### **4.2.2.2 Visual Studio with Web Camera**

We decided to use the web camera with a visual studio rather than MATLAB. Visual Studio is a programming environment which used to enter the OpenCV library.

The initial procedure began with capturing the signer video by the web camera in the real time. The next step should be the skin detection but we faced many problems such as the complex background problem that merge with the skin color of the signer, because of this reason we faced difficulty to make this step and the segmentation step. We resolved this problem by making Region of Interest (ROI) which is a square with certain dimensions that the signer put his hand inside it and the whole process execution on it (Figure 4.1).

The ROI just resolve the skin detection problem but the segmentation problem still existing.

Through this method, we have been able to obtain results are fairly satisfactory for a detection and weakened results of tracking. The web camera has more limitations such as; it need high and powerful light, the background of the signer should have a certain distinguished color, sometimes the signer should wear gloves with a certain color, and also it is difficult to differentiate between two persons appear on the screen. All of these restrictions have made us look for an alternative to the web camera and after searching we found Kinect camera.



Figure 4.1: Region of Interest

#### **4.2.2.3 Visual Studio with Kinect Camera**

Microsoft Kinect was initially developed as a peripheral device for use is XBOX 360TM gaming console. It has three sensors (RGB, audio, and depth). It can provide "color frames, depth data and skeleton data" besides also capturing audio stream that may be used in voice recognition. It was much cheaper and easier to use than any other 3D imagers, and it made 3D imaging accessible to many more researchers. Microsoft also provided a very high-quality Software Development Kit (SDK) for developing gesture-based applications. As a result of this, the Kinect was -and still is- very popular. So for these reasons, we preferred work with it.

The methodology we initially set up to make sign language recognition into Arabic letters by using Kinect camera is shown in figure (4.2).



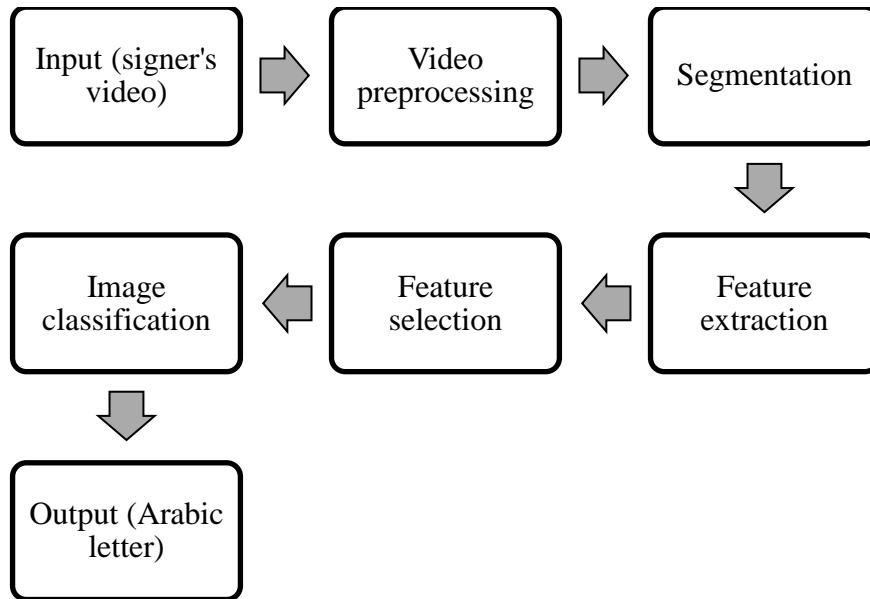


Figure 4.2: block diagram of initial recognition ArSL to Arabic letters system

The initial recognition system begins with capturing of video, then some preprocessing have been done on video like noise reduction and contrast enhancement, then it entered to segmentation process. This process contains two stages shown in figure (4.3). The first stage is a singer segmentation process, the purpose of it to an isolation of the singer body from the whole environment as this process is followed obtain the closest person to Kinect sensor, get the binary depth image of this person and Mask color image with the binary depth image. The second stage is hand segmentation process, the main target of it to segment the signer's right hand, after signer segmentation the skin segmentation applied by using the RGB ratio model then get the x, y coordinates of the right hand and left hand using skeleton data and in case of both hands detection, after that constructed an image containing right-hand and left-hand contours flood-filled with white color by getting contours containing these x, y coordinates from Kinect SDK then obtain only the right and left hands by Mask signer-segmented image with binary hands. Finally, focus on right hand only, extract right-hand rectangle and convert it to gray scale. Then feature extraction process can be done by using some algorithms. After feature-extraction, it is time to classification these features by using machine learning techniques and recognizes the signs into Arabic letter.

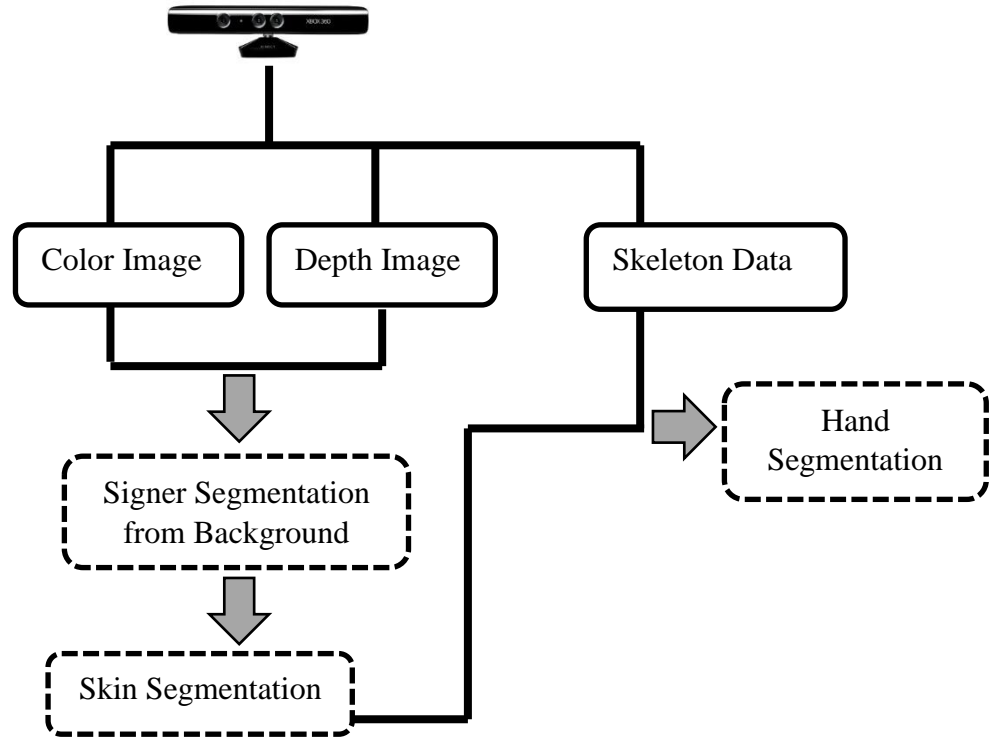


Figure 4.3: Overall Segmentation Process design

We could not work the hand segmentation because we had difficulties in Most of the steps. These difficulties started with hand complexity and difficulty, so the accurately model of the hand joints is difficult to design. Another problem is the color stability and the challenge of hand fragmentation from its background, and most of the hand gestures in Arabic letters are similar; therefore it needs to be very accurate in calculating the dimensions of the joints which requires the design of new algorithms to obtain skeleton information of hand joints from a Kinect camera. As a result, we decided to follow a new method of solution, which is to make tracking of the skeleton joints and thus have been benefiting from the features that provided by the Kinect. This method enables us to get words and phrases instead of letters.

### 4.3 The Proposed System

The way it was followed to implement the goal of Arabic sign language recognition is divided into 4 basic stages as shown in the figure (4.4)

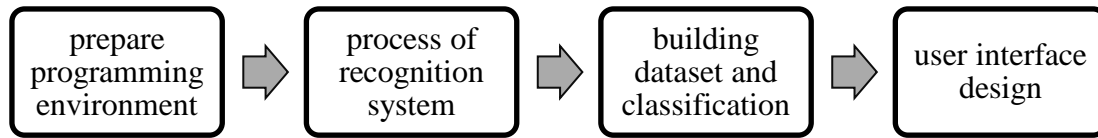


Figure 4.4: four basic stages of proposed system

### **4.3.1 Preparing the Programming Environment**

The visual studio programming environment has been prepared by install C# language support on it, selects the Windows Presentation Foundation (WPF) platform to build Windows desktop application, install the Kinect SDK and developer toolkit, and entered the Kinect sensor as a reference in the visual studio.

### **4.3.2 Process of ArSL Recognition System**

The recognition system is consisting of several phases. These stages are started with video capturing by Kinect camera, obtained depth image, and obtained skeleton data, finally feature extraction.

#### **4.3.2.1 Capture the signer's video**

As shown in figure (4.5), The Kinect sensor is built-in color camera, infrared (IR) emitter, and microphone array. RGB camera that stores three channel data (Red, Green, and Blue) used to provide 2D color frames and capture the signer's video as real-time video with high accuracy that appears directly in visual studio. All the upcoming processing was done in the same time as the camera captures the real-time video.

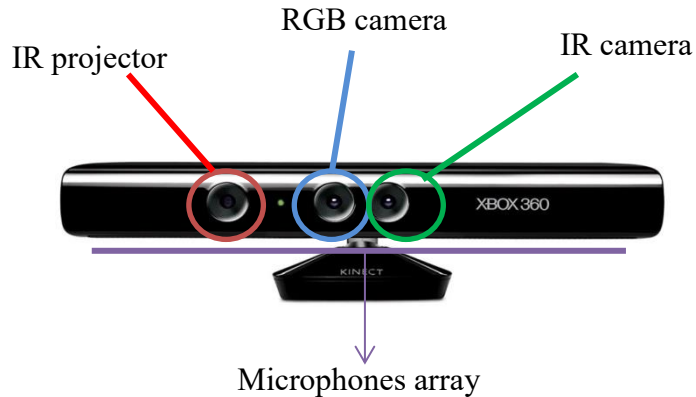


Figure 4.5: kinect sensor hardware

#### 4.3.2.2 Obtain the Depth Image

One of the basic components of kinect sensor is an Infrared (IR) emitter and an IR depth sensor. The emitter emits infrared light beams and the depth sensor reads the IR beams reflected back to the sensor. The reflected beams are converted into depth information measuring the distance between an object and the sensor. This makes capturing a depth image possible (figure 4.6).

The basic depth data is important in building any really useful Kinect application. It gives more than a basic image that tells how far away each pixel in the image is. It can take the depth field and label each pixel with which "signer" it is part of as well as performing a complete skeletonization to show where the signer limbs are.

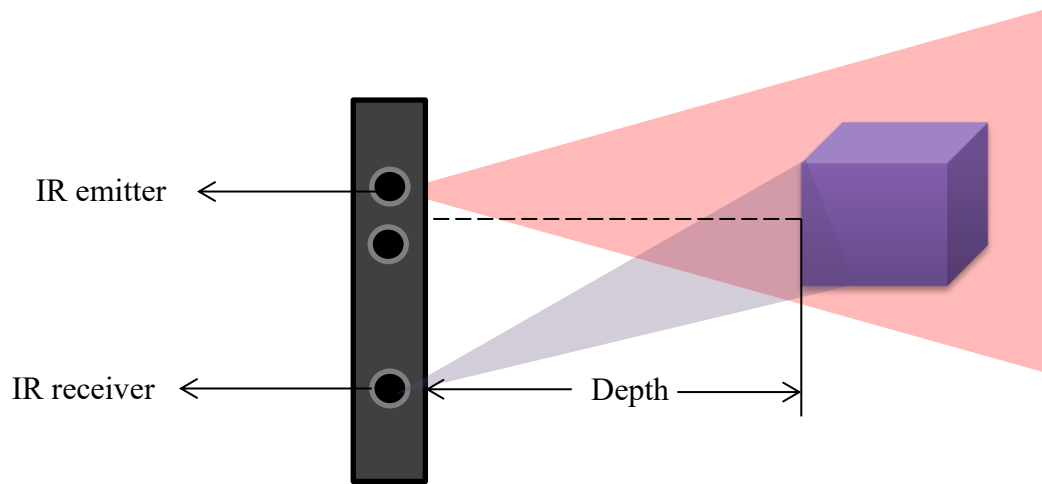


Figure 4.6: obtain the depth information of the object by the kinect sensor.

Depth frame is scanned and pixels that represent the nearest objects to the Kinect sensor are labeled as 65,535 for white. By depth frames, the objects details have been reduced so the frames are ready for next process phase.

#### 4.3.2.3 Skeletonize Data

Stream data is delivered as a succession of still-image frames. Detect of the skeleton from depth frame is a classification method that done by one of the most important algorithms of machine learning is a Randomization Decision Forest algorithm. Kinect using this algorithm to inferred the body part position. This algorithm is learned from 1 million depth image with known skeletons by motion capture system. Randomized decision forest is a more sophisticated version of the classic decision tree.

Kinect uses the mean shift algorithm that is a simple, fast, and effective algorithm which transforms the body part position -that detected by the previous algorithm- into a skeleton. Figure (4.7) illustrates the skeletonize data phase.

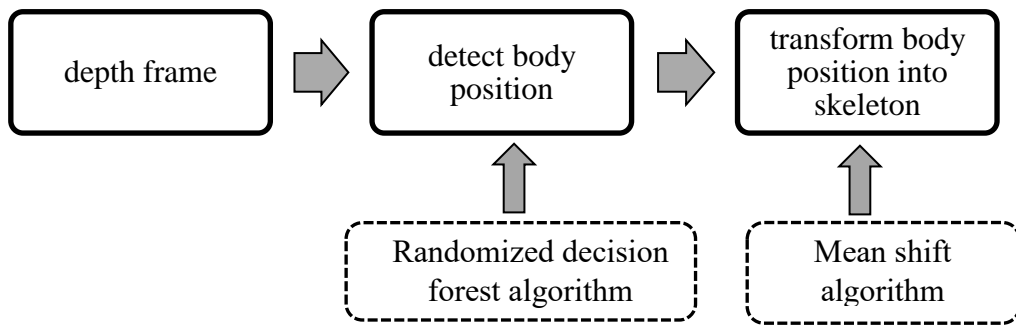


Figure 4.7: the skeletonized data phase

The Natural User Interface (NUI) Skeleton API provides full information about the location of up to two users standing in front of the Kinect sensor, with detailed position and orientation information. The data is provided to an application as a set of endpoints, called skeleton positions that compose a skeleton.

A Joint position returns X,Y,Z values as explained below (figure 4.8)

- X = Horizontal position between -1 and +1
- Y = Vertical position between -1 and +1
- Z = Distance from Kinect measured in meters

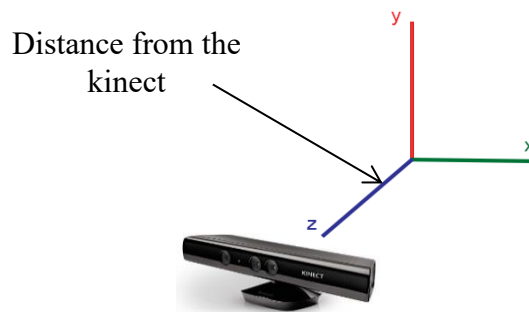


Figure 4.8: XYZ plane that describe joint position

In this phase, we used the NUI Skeleton Data Structure array that contains the data for one skeleton, including overall position, skeleton joints positions, and whether each skeleton joint is tracked successfully.

Another array used is a Skeleton Position array which contains the position of each joint. The following figure (4.9) shows the order of the joints returned by the Kinect adaptor. When Body Posture is set to Standing, all 20 indices are returned.

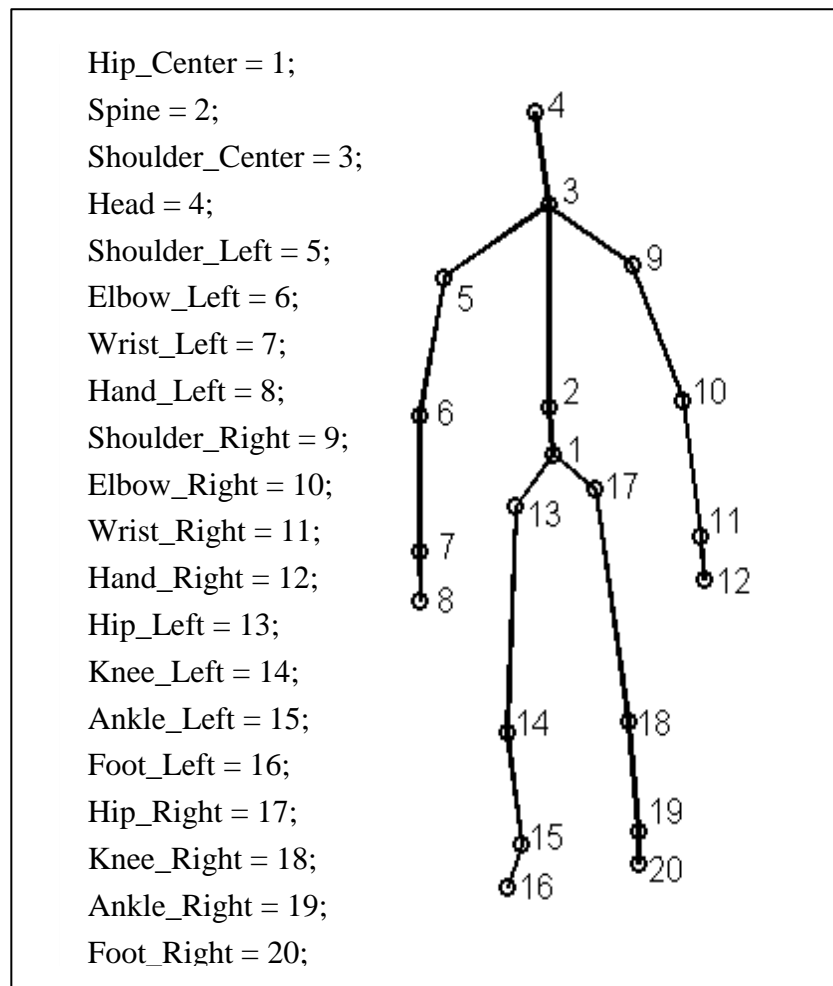


Figure 4.9: positions of 20 joints in the skeleton

#### 4.3.2.4 Skeleton Tracking

The information about recognized signer is provided as an array of skeleton objects and dealing with it as a frame.

Skeleton frame has two options of tracking it can be in tracked state or position only state. The tracked state gives detailed information about the position of all 20 joints

every time but the position only state gives information about the position of the signer so we prefer to use the tracked option.

Kinect SDK has lots of features that help in ArSL recognition process. One of them is Skeletal Tracking. To track the signer's skeleton movement, the application needs to enable skeletal tracking feature by using the special method and write it in C# code. This method called (SkeletonStream.Enable ) method. For skeleton tracking, the user should stand between 4 and 11 feet from the sensor.

#### 4.3.2.5 Feature Extraction

In the machine learning, features are special attributes that defined quantitatively. Feature extraction is the process that quantifying the attributes and by it is obtained a sequence of numbers related to relevant information about the attribute and useful for the next classification stage.

Feature extraction is the most important stage in the process of ArSL recognition because it determined the quality of the next classification stage.

Sign language always involves the upper body movement; therefore feature calculation takes into account the upper joint movement which is represented in the head, elbow, shoulder, spine, and hand orientation. The other joints are ignored.

Consider the set of coordinates is S which  $|S| = 8$ .

The Number 8 represent the coordination of head, spine, right shoulder, left shoulder, right elbow, left elbow, right hand, and left hand.

Then computed the 3D centroid coordinate by the equation:

$$\vec{C} = \frac{\text{shoulder(right)} + \text{shoulder(left)}}{2} \quad (4.1)$$

All remaining coordinates are normalized by subtracting the centroid:

$$\forall \vec{F} \in S, \quad \vec{F}' = \vec{F} - \vec{C} \quad (4.2)$$

Where  $\vec{F}$  is the coordinate of any joint from the set S.

Then the distance between right and left shoulder (D) calculated, and all coordinates further normalized by dividing by the shoulder distance.



$$\forall \vec{F}' \in S', \quad \vec{F}'' = \frac{\vec{F}'}{D} \quad (4.3)$$

This further makes the algorithm Scale-Invariant Feature Transform.

### 4.3.3 Building Database and Classification

The Kinect Stream Application allows users to display and store the Kinect streams like skeleton stream. This application is developed based on a sample called "Kinect Explorer - D2D C" developed by Microsoft Corporation.

The depth images are saved as single channel short unsigned integer images. This is the format by the Kinect which can read this by using OpenCV function.

The dataset which was building is contained a set of depth frames and features of skeleton joints that extracted from the previous stage.

The dataset used in this system is consist of 6 words in ArSL namely "مرحباً", "جائع", "عمر", "ماذا", "أنت", "اسم"

#### 4.3.3.1 Classification

The Dynamic Time Warping (DTW) is an algorithm that used to calculate the optimal matching between the real-time video that Kinect is capturing and the gesture from the dataset of gesture sequences.

The classification method is a comparing between features that extracted from the real-time video with the feature that saved in the dataset and related to the specific word. If the two features are matched, the related word appears as the intended word. (See figure 4.10).

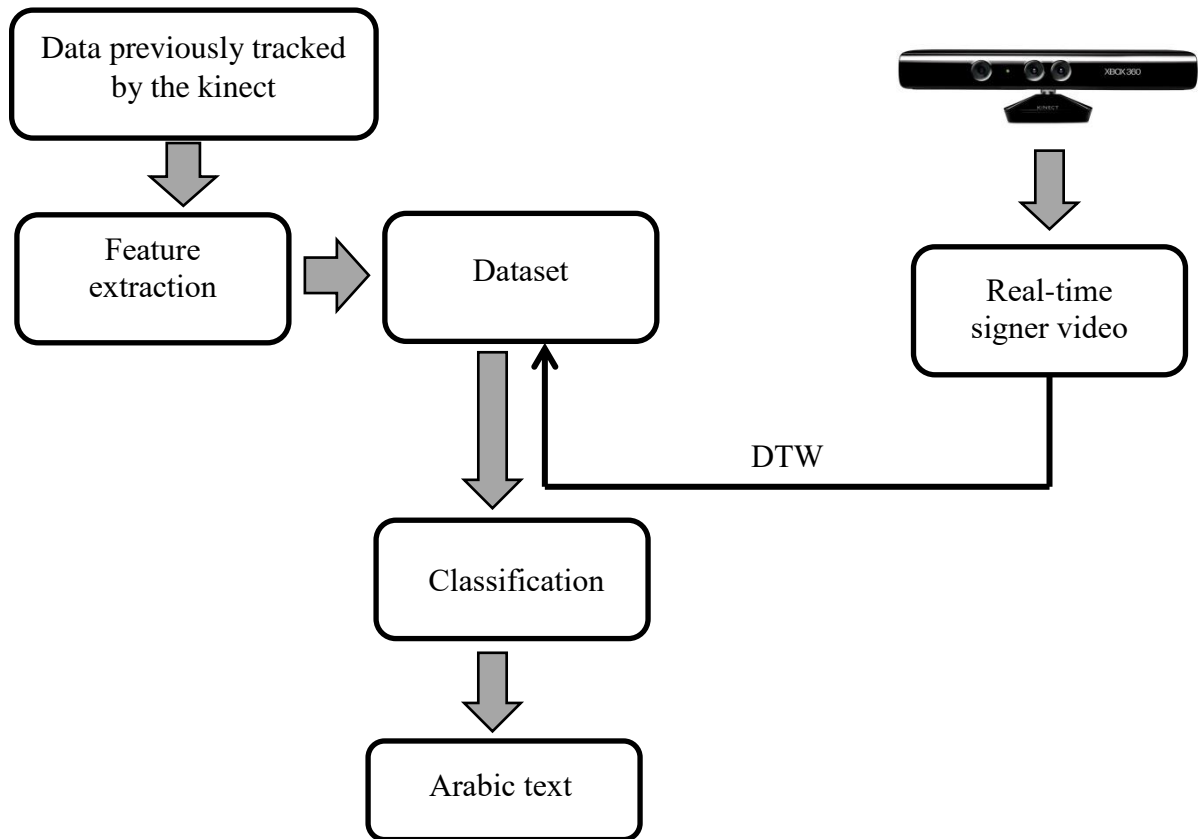


Figure 4.10 describe matching real-time video with features in dataset

In sign language, the sequence of words in sentences is different in normal speech. so that to prevent the sign language recognition to be a literal translation we put an estimating process to the intended sentences such as if the system finds two words ("اسم", "انت") it will estimates the sentence to be "ما اسمك؟".

**Table 4.1:** illustrates the sentences used in this system

<b>Required Words</b>	<b>Estimated Sentence</b>
اسم , أنت	ما اسمك؟
عمر , أنت	كم عمرك؟
جائع , أنت	هل أنت جائع؟

#### **4.3.4 User Interface Design**

The goal of user interface design is to make the system easily when used. The simple design is illustrated in the figure (4.11)

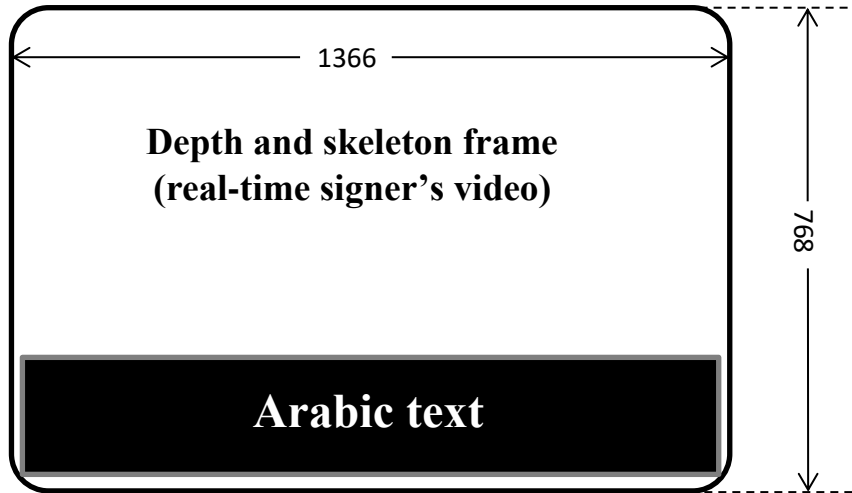


Figure 4.11 user interface design

# **CHAPTER FIVE**

## **RESULTS AND DISCUSSION**

This chapter presents the project outcomes. In Section 5.1 the dataset used in the experiments is introduced. In Section 5.2, the environment factors which that we used. In Section 5.3, Illustration of the result obtained for the depth image and the skeletal tracking. In Section 5.4, the experiments results. In section 5.5, the Number of Attempts through which the wanted result is obtained. In section 5.6, the Time of System Response.

### **5.1 Dataset**

The dataset consists of Arabic Sign Language signs that captured by using Kinect sensor. The next step after the capturing video is tracking the movement of the skeleton joints. Then make feature extraction by using some algorithms. These quantitate features are stored as single channel short unsigned integer images. This is the format by the Kinect which can read this by using OpenCV function. SDK framework is used for skeleton tracking and the dataset contains 6 signs.

### **5.2 The Environment Factors**

The data collection was done in a room with white light. The person position at 180 cm away from the front of the Kinect Xbox 360 device. The Kinect was mounted on a table 60 cm from the floor.

### **5.3 Depth Image and Skeletal Tracking**

The image in (Figure 5.1) has shown the depth image that obtains from the Kinect sensor. The Kinect sensor provides depth information that through which we were able to make detection of the skeleton and tracking of the movement.



**Figure 5.1: Depth image**

The depth image reduces the details in the colored scene for that the skeleton detection process and tracking process becoming more easy and accurate.

The image in (Figure 5.2) showed the skeletal detection and tracking that obtain from the Kinect sensor by using two algorithms Randomized decision forest algorithm and Mean shift algorithm.

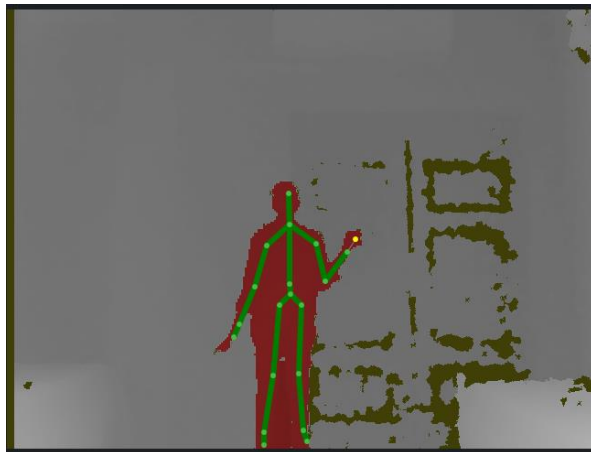


Figure 5.2: skeletal detection and tracking

From the tracking of skeleton movement, we were able to make recognition on the sign.

## 5.4 Experiments Results

Six different Arabic SL gestures and three sentences were tested

”اسم”, ”جائع”, ”عمر”, ”انت”, ”ماذا”, ”مرحبا”

”هل انت جائع?”, ”ما اسمك?”, ”كم عمرك?”

All following figures illustrate the result of system recognition of these gestures on dataset.

The gesture in figure (5.3) involved putting the right hand joint in the same position of the head joint. Right-hand joint also should be above shoulder center and left of the right shoulder.



Figure 5.3: gesture recognition of "مرحبا"

The gesture in figure (5.4) involved putting the right hand joint and the left hand joint above the left and right shoulder joints.



Figure 5.4: gesture recognition of "ماذا"

The gesture in figure (5.5) involved putting the right hand joint above the left elbow joint.

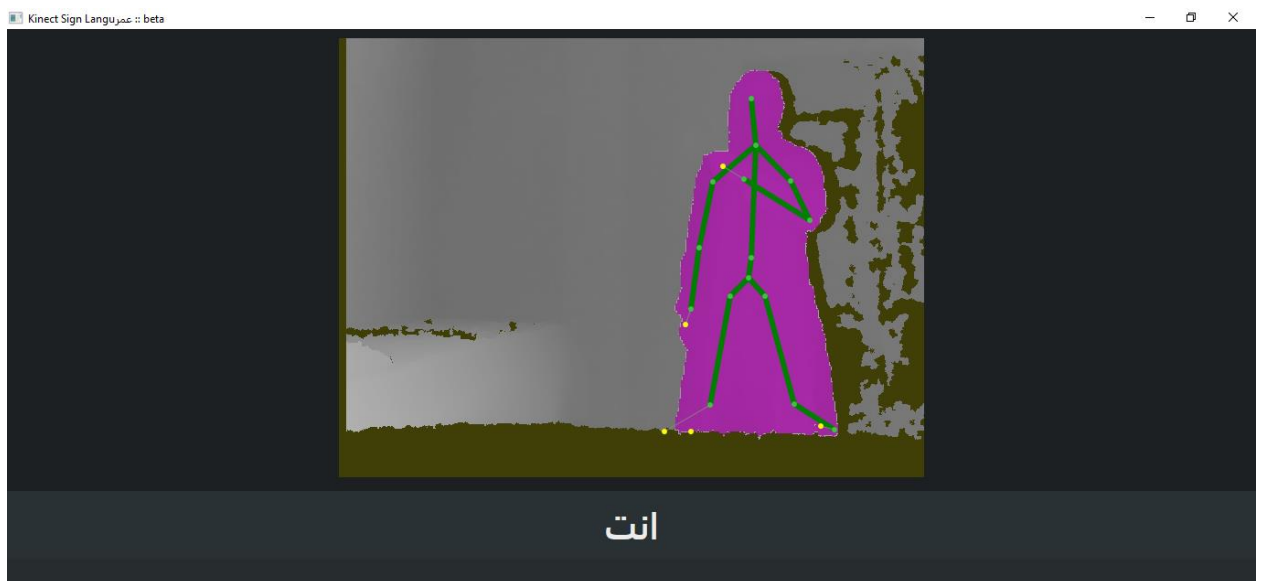


Figure 5.5: gesture recognition of "انت"

The gesture in figure (5.6) involved putting the right hand joint between the head and the center of shoulders

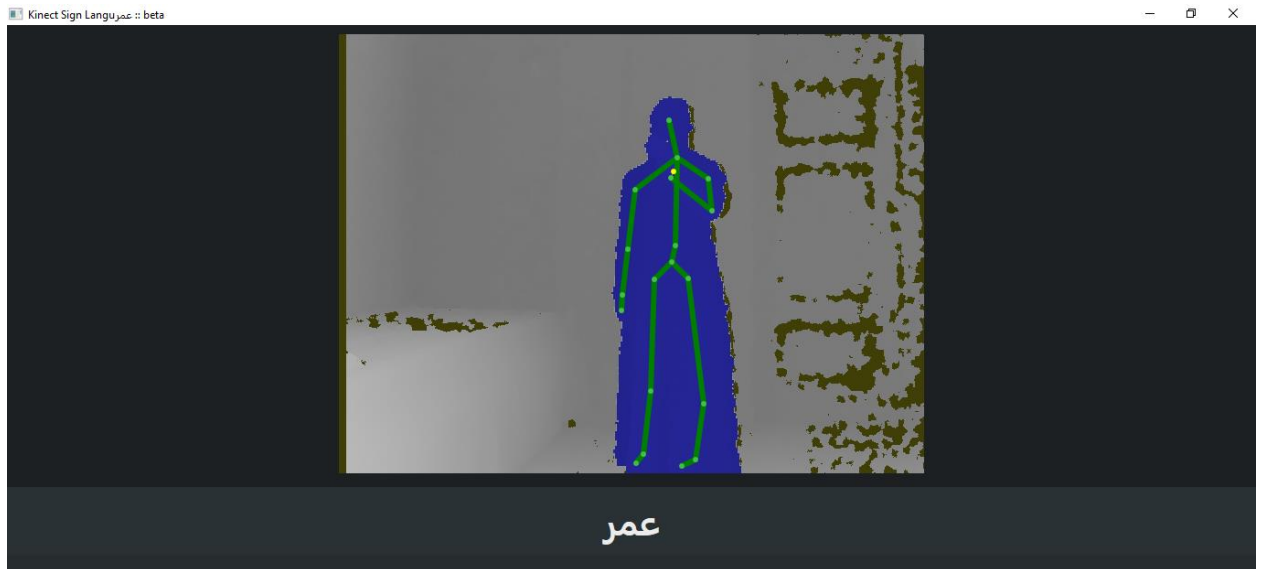


Figure 5.6: gesture recognition of "عمر"

The gesture in figure (5.7) involved moving the right hand from the spine to right and left hip joints up and down.



Figure 5.7: gesture recognition of "جائع"



The gesture in figure (5.8) involved both right and left hand joints should be in the center of the shoulders and in the position of the spine.

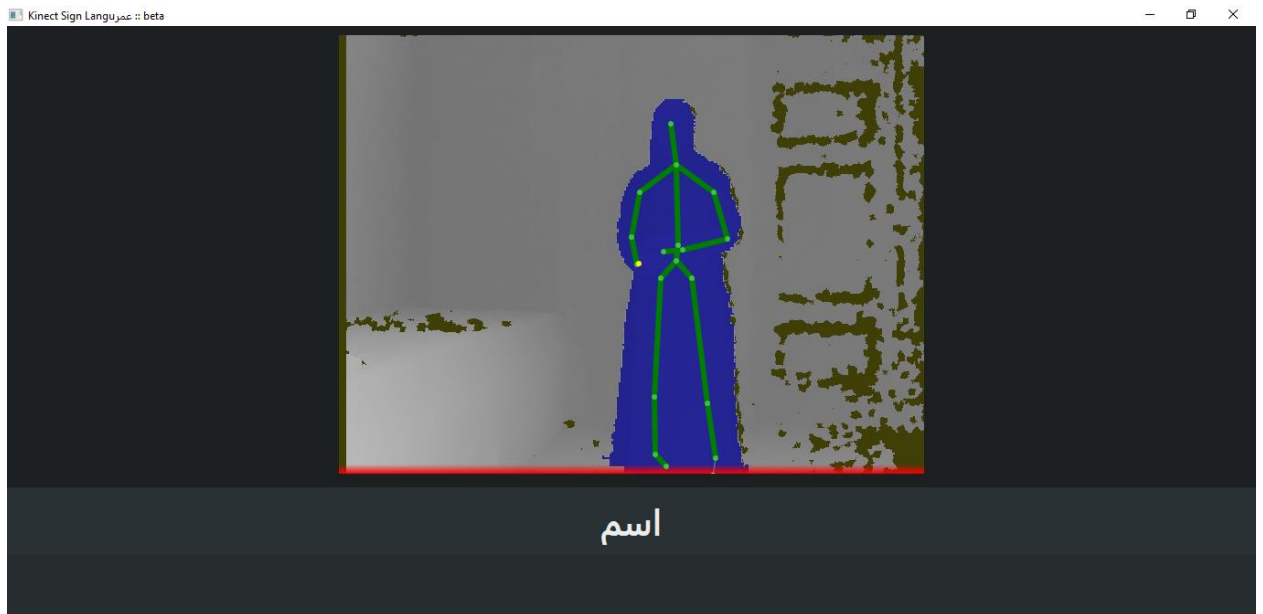


Figure 5.8: gesture recognition of "اسم"

**Note:** The response in Figure 5.3, Figure 5.4 and Figure 5.5 it is at the real-time but the response in other Figures delay a few second because the recognition in real-time need high specification processor and our personal computer is low specification so the response of this Figures is delay a few second.

#### 5.4.1 Build Up The Sentences

Build up the sentences in this system achieved by estimating the sentence from defined words appears with a specific sequence.

The sentence of figure (5.9) will be building when the gestures occurs in the sequence of "عمر" and then "أنت"



Figure 5.9: gesture recognition of "كم عمرك؟"

The sentence of figure (5.10) will be building when the gestures occurs in the sequence of "انت" and then "اسم"

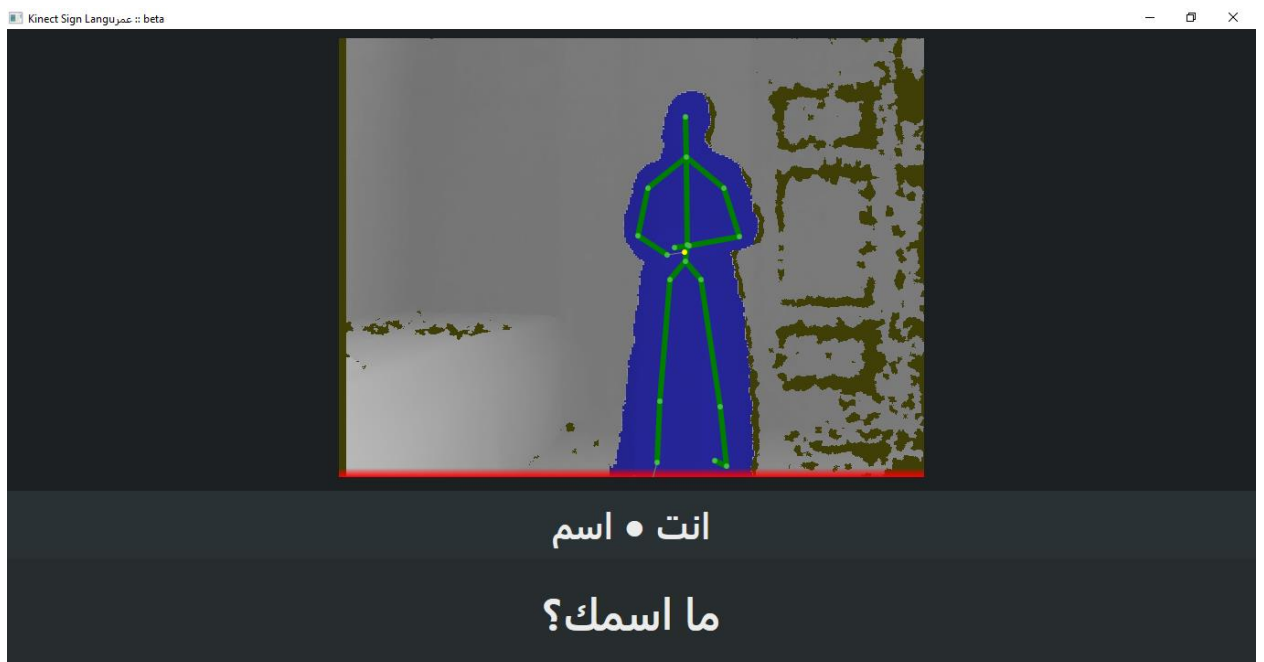


Figure 5.10 :gesture recognition of " ما اسمك؟ "

The sentence of figure (5.11) will be building when the gestures occurs in the sequence of "أنت" and then "جائع"

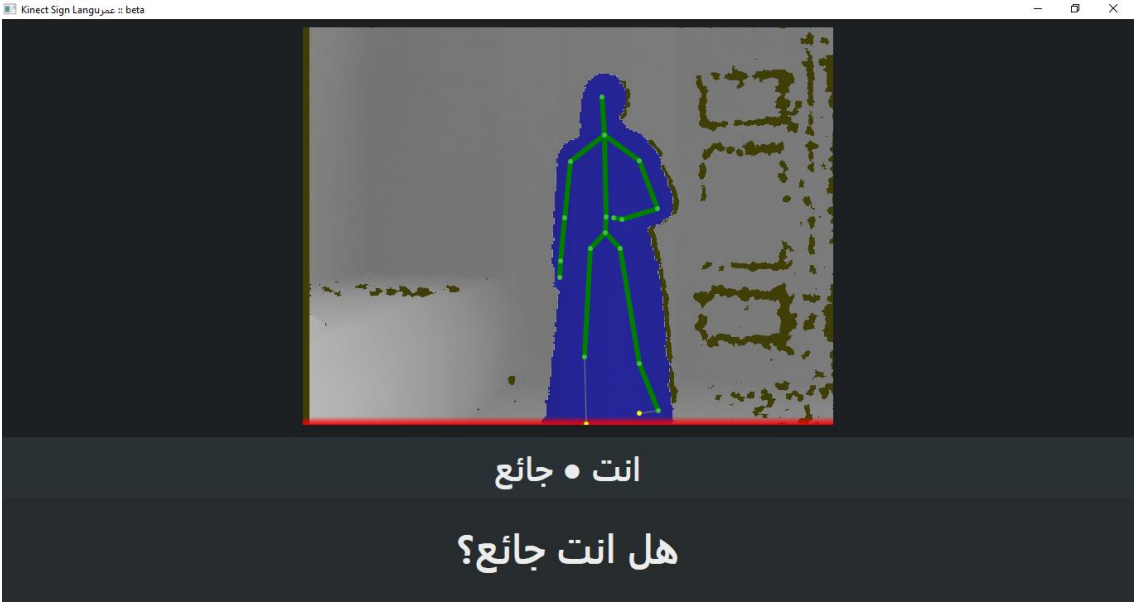


Figure 5.11: gesture recognition of "هل انت جائع؟"

If more than one user are stand front the Kinect camera, the kinect just response to nearest user and do the process of recognition just to him. figure (5.12)

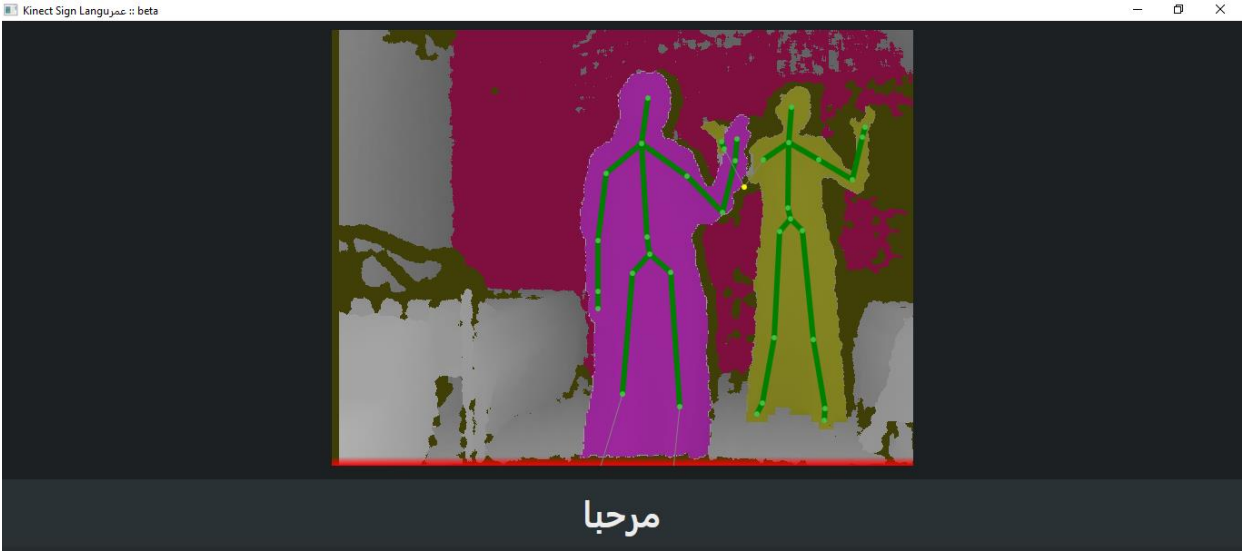

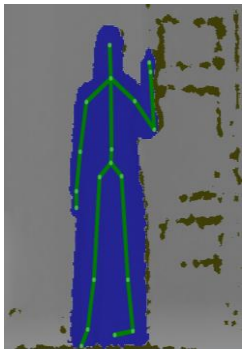

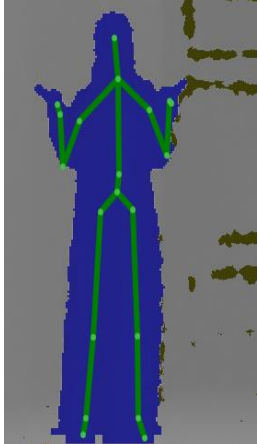

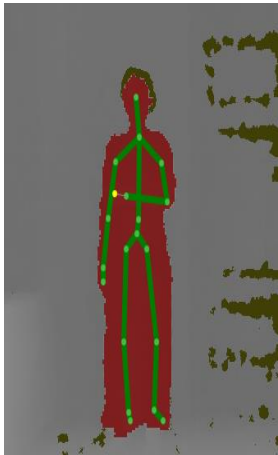

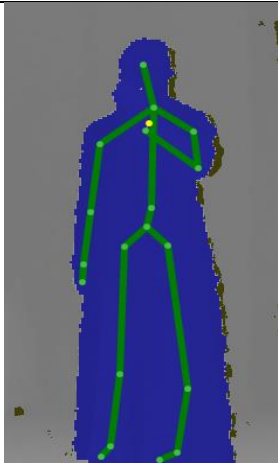


Figure 5.12: shown the Kinect sensor is just response to the close person.


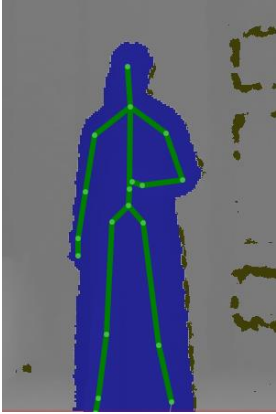

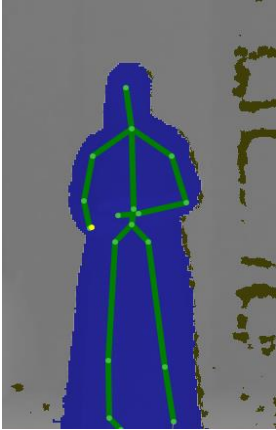
Table 5.1: Skeleton frame of different signs.

Sign	Image	Skeleton Understanding	Users	Result
مرحبا			1	Pass
ماذا			1	Pass

Continue: Table 5.1: Skeleton frame of different signs.

Sign	Image	Skeleton Understanding	Users	Result
انت			1	Pass
عمر			1	Pass

Continue: Table 5.1: Skeleton frame of different signs.

Sign	Image	Skeleton Understanding	Users	Result
جائع			1	Pass
اسم			1	Pass

## 5.5 Number of Attempts

After 5 times experiments for each sign, we found that some signs the system does not response to it from the first time. The maximum number of attempts for each sign is illustrated in the following table (5.2)

Table 5.2: The maximum number of attempts which system responds to the sign

Sign	Maximum No. of attempts
مرحبا	1
ماذا	1
انت	1
عمر	2
جائع	3
اسم	2

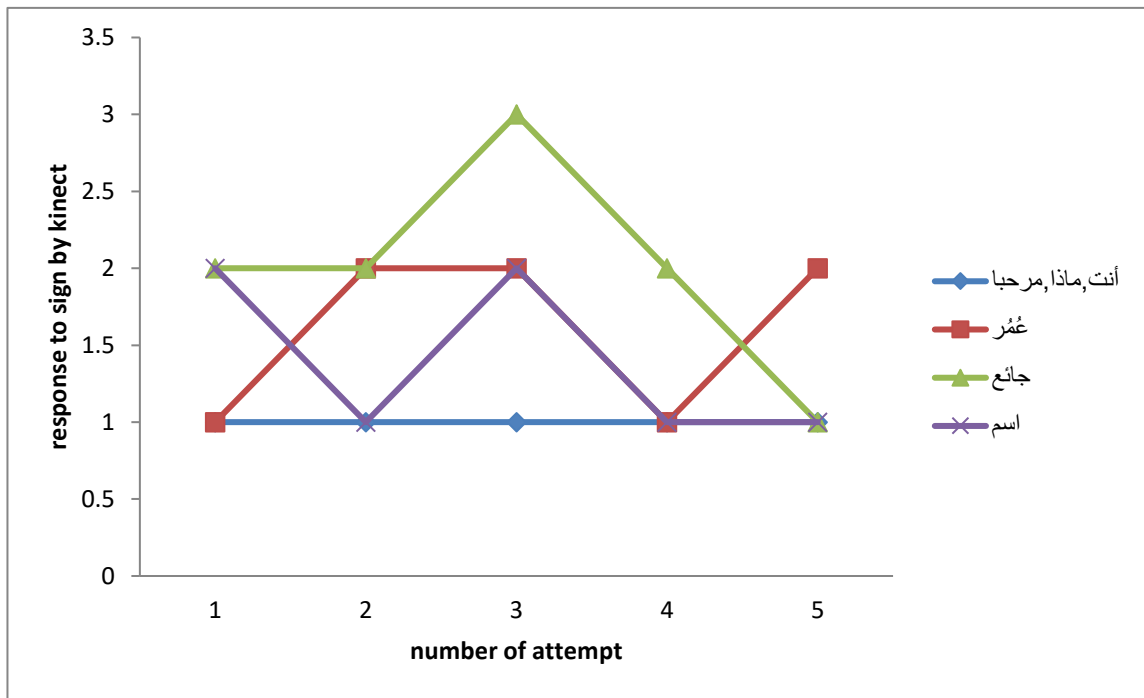


Figure 5.13: the number of attempts which system responds to each sign

## 5.6 Time of Response

Kinect camera capturing the signer who is the person standing front of it and then all the process of recognition the gestures of signer are should be done in real time but in this system, it takes specific time shown in the table (5.3). For calculating the average time which takes the camera to recognition, the experiment of doing gestures was repeated five times, and then the average was taken.

Table (5.3) the time of response of the system for each sign

Sign	Time of response (By seconds)					Avg. (By sec.)
	1	2	3	4	5	
مرحبا	0.55	0.57	0.59	0.48	0.50	0.54
ماذا	0.55	0.43	0.39	0.57	0.30	0.45
انت	0.7	0.95	0.79	0.78	0.31	0.71
عمر	1	1.04	0.82	0.96	0.70	0.9
جائع	1	1.2	1.18	1.22	1.25	1.17
اسم	3.48	3.37	3.20	3.21	3	3.25



The following figures show the time of response for each sign.

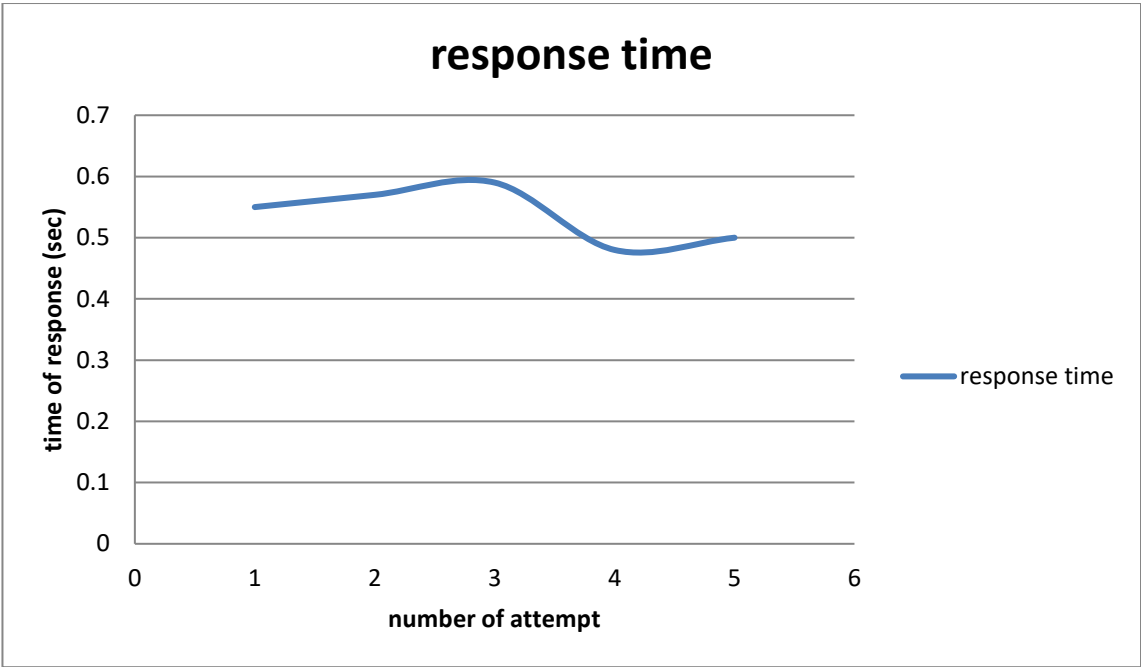


Figure 5.14: time response of "مرحبا" sign.

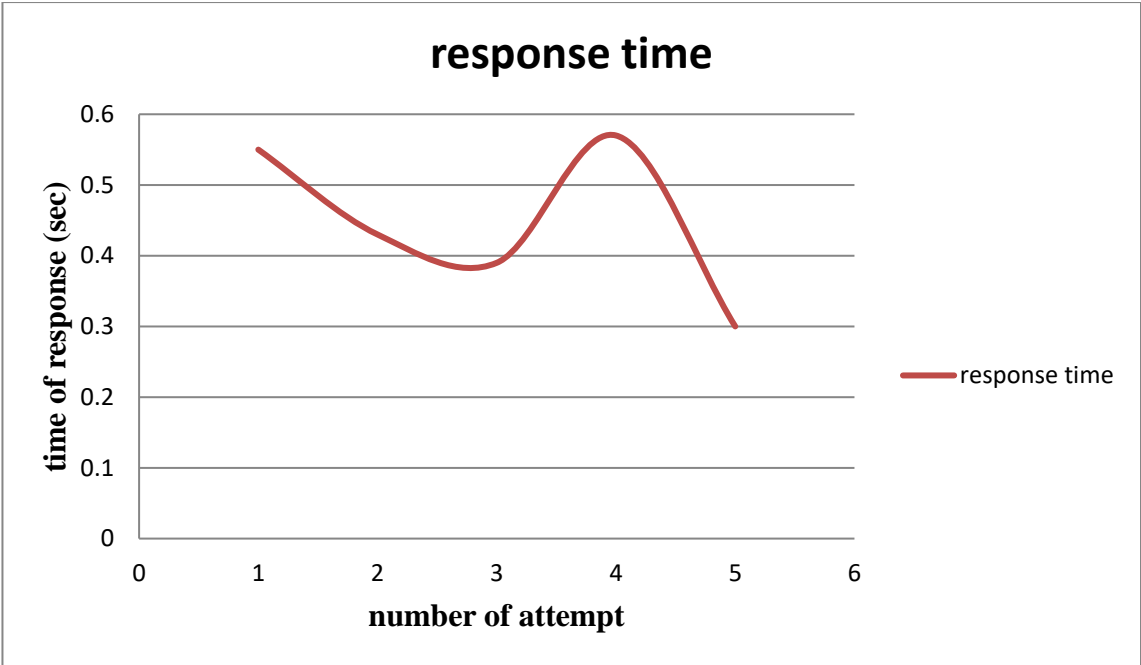


Figure 5.15: time response of "ماذا" sign.

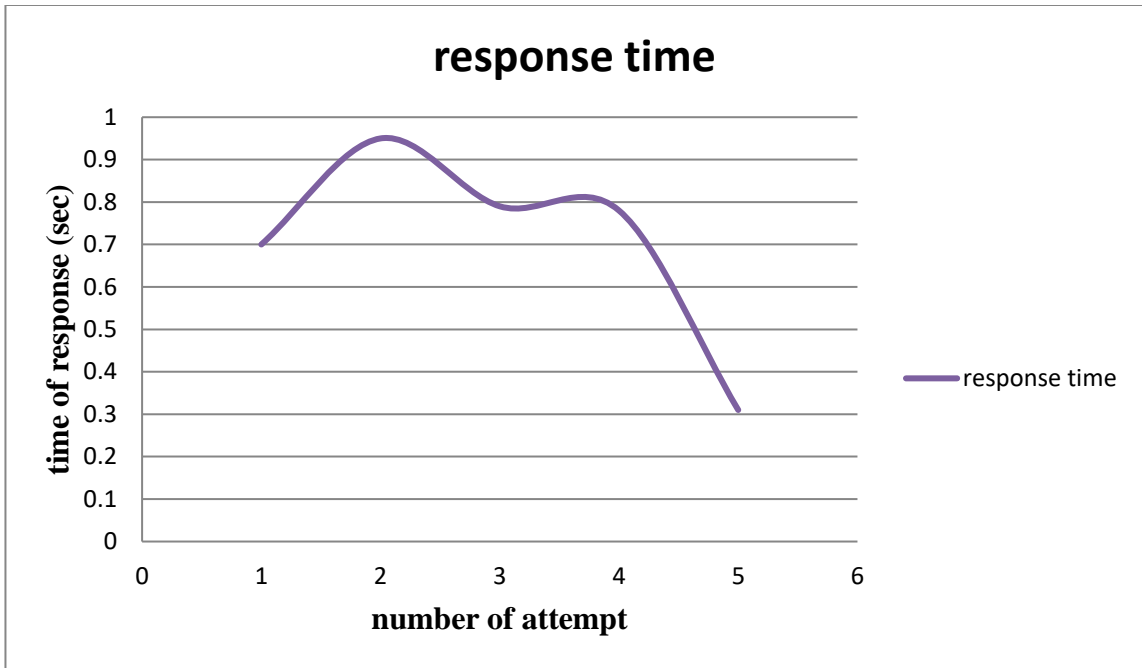


Figure 5.16: time response of "أنت" sign.

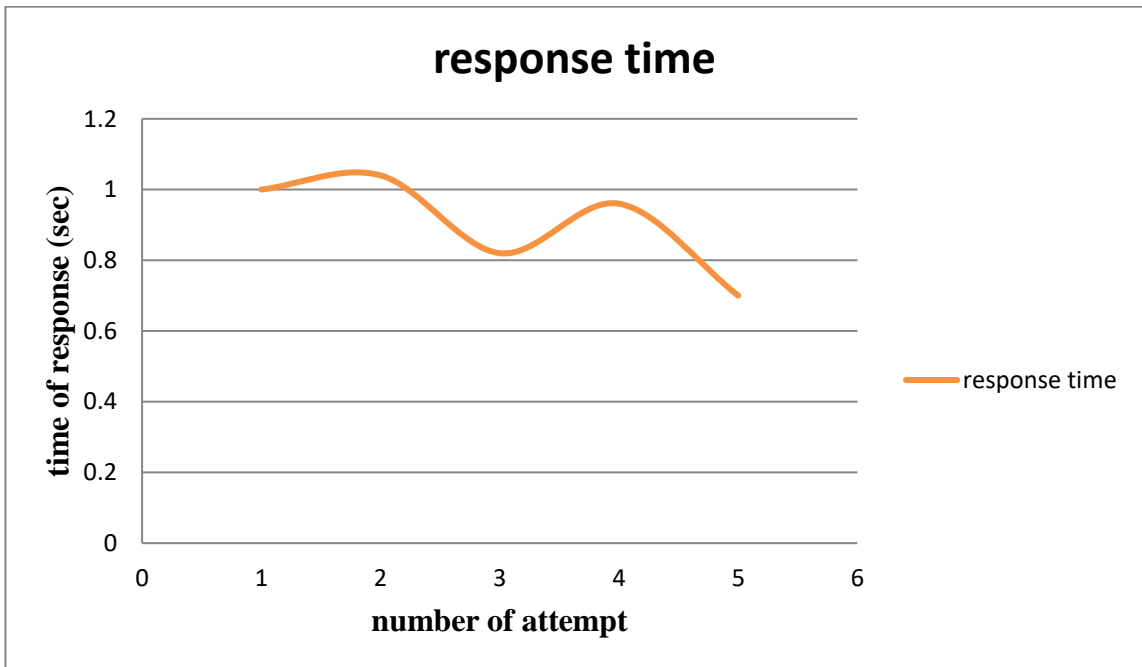


Figure 5.17: time response of "عُمر" sign.

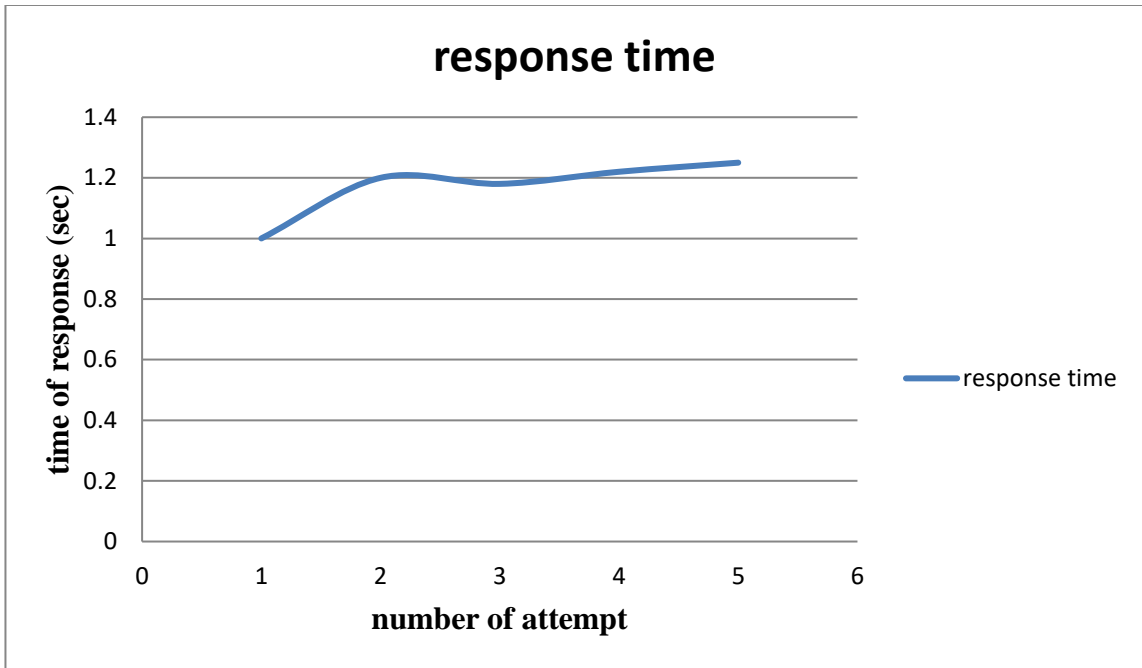


Figure 5.18: time response of "جانع" sign.

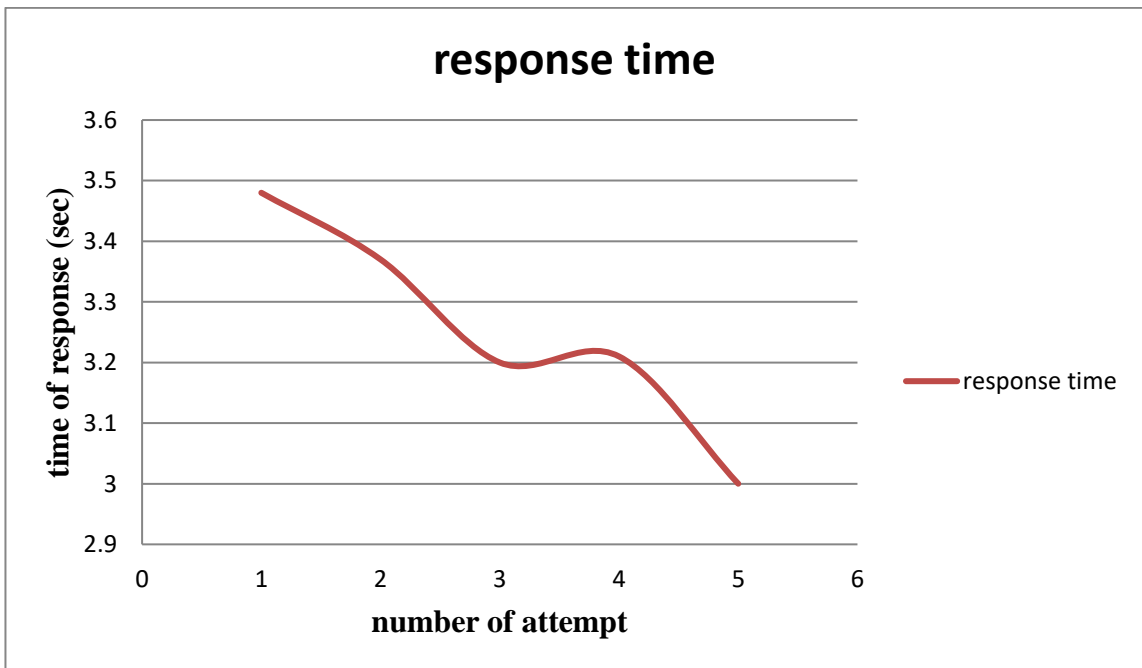


Figure 5.19: time response of "اسم" sign.

The response time values for each individual sign are somewhat similar, that indicating the system is precision. And the values range from 0.3 fragment of second and 3 seconds which indicating that the system has fast response.

# CHAPTER SIX

## CONCLUSION AND FUTURE WORK

In this chapter, a summary of the proposed method is given and it is assessed by considering its strengths and weaknesses. The Section 6.1, contain summarized and advantages of proposed system. The Section 6.2, contain limitation and what can be done in the future work.

### **6.1 Summary and Advantages**

This thesis presented a novel approach for Arabic SLR by using the Kinect sensor. The Using of Kinect sensor to capture sign gestures is more useful and simpler environmental setup compared to the systems using data gloves or Combination of cameras. Kinect is a ready-to-use sensor which can detect users without the need of explicit calibration and start tracking immediately whereas other sensors that require complex environmental setup and calibration. In this manner, use of Kinect sensor makes the proposed method advantageous in terms of system setup and mobility. The proposed system also gets benefit of Kinect in terms of accuracy. Through its embedded depth camera, color camera and infrared sensor, Kinect provide locations of skeletal joints accurately. Another benefit of using Kinect is real-time tracking of body joints.

In the scope of this thesis, the system that able to recognition for six words and three sentences was designed. That process has been done in several steps and these steps initial by getting the depth information from the Kinect Sensor and through this information, detection of the skeleton and tracking of the movement was achieved, after that feature extraction and classification process has been done.

The proposed method employs Kinect camera that has a lot of advantages that make it the best of other depth cameras and it is easy to programming and more accurate, further it is a low-cost tool. In this proposed method the algorithms and methods are easiest to use and fastest performance. The algorithms that used for classification DTW algorithm. The advantages of it allow faster and more accurate classification.

Finally, the important point is that the proposed method shows that recognition of the Arabic sign language it can become possible in Sudan because it is available at the lowest cost.

## **6.2 Limitations and Future Work**

The recognition process needs a high-specification device and efficiently handled so the first obstacle we encountered is the limitation of the processor in our personal computer to conduct the process quickly and efficiently.

The parameters of the hand that take into account in recognition system are; hand shape, movement, location, orientation. Language has hundreds of signs, some of which differ from each other in only one of these parameters, for example, some movements in the Arabic sign language have the same location and orientation but differ in the shape of the hand and other movements depend on the shape of finger and hand. Also there are various word have the same signs but differ in the face expressions so we had trouble building the code includes all Arabic words.

Through the observing the limitations of the proposed solution, it is possible to identify multiple future work directions. These include:

- Make a hand joint segmentation in order to add Arabic letters and make face recognition in order to obtain the face expressions and increase the database to include all the vocabulary of the Arabic sign language.
- Add an option to the program so that it converts the speech into the Arabic sign language and also translates the sign language into other sign languages or the Reverse
- Applied this proposed solution in the field of health, education and public institution