Real-time glove and android application for visual and audible Arabic sign language translation

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 163 (2019) 450–459

Procedia Computer Science

www.elsevier.com/locate/procedia

16th International Learning & Technology Conference 2019

Real-time glove and android application for visual and audible Arabic sign language translation

Nema Salem[a*], Saja Alharbi[b], Raghdah Khezendar[c], Hedaih Alshami[d]

[a]Assistant Professor, Electrical and Computer Engineering Department, Effat University, Jeddah 21478, KSA
[b, c, d]Undergraduate students, Electrical and Computer Engineering Department, Effat University, Jeddah 21478, KSA

Abstract

Researchers can develop new systems to capture, analyze, recognize, memorize and interpret hand gestures with machine learning and sensors. Acoustic communication is a way to convey human opinions, feelings, messages, and information. Deaf and mute individuals communicate using sign language that is not understandable by everyone. Unfortunately, they face extreme difficulty in conveying their messages to others. To facilitate the communication between deaf/mute individuals and normal people, we propose a real-time prototype using a customized glove equipped with five flex and one-accelerometer sensors. These sensors are able to detect the bindings of the fingers and the movements of the hand. In addition, we developed an android mobile application to recognize the captured Arabic Sign Language (ArSL) gestures and translate them into displayed texts and audible sounds. The developed prototype is accurate, low cost and fast in response.

Keywords: Sensory glove; Arabic Sign Language; Flex sensor, Accelerometer sensor; MIT inventor; Mobile application; Deaf and mute

1. Introduction

Communication is an act of conveying and exchanging information that can be accomplished efficiently only when all the participants use a common language for interaction. Hand gestures are used for interaction with the environment around us in a massive number of tasks in our day-to-day lives. Communication through gestures is assessed as warm,

* Corresponding author. Tel.: +966920003331; fax: +966126377447.
E-mail address: nsalem@effatuniversity.edu.sa

10.1016/j.procs.2019.12.128

friendly and active, whereas communication with no gestures encompassed with it is viewed as logical, emotionless, and diagnostic.

According to the World Health Organization, over 5% of the world's population that about 466 million people has a disabling hearing loss (432 million adults and 34 million children). By 2050, over 900 million people will have a disabling hearing loss. In addition, about 10 million people in the Middle East are deaf and mute. In Saudi Arabia, the amount of mute people is about 800 thousand people [1].

Deaf and mute people are facing difficulties in understanding and communicating with normal people. It is important that researchers solve this problem, so deaf and mute people can communicate with normal people, express their emotions and convey their opinions, and not feel distanced or lonely. Moreover, they do not feel helpless and misunderstood by the normal people. It gives them career opportunities and is beneficial for companies in providing labor.

This research design and implement a sensory glove that can effectively recognize diverse hand Arabic letters. The developed glove consists of two main parts. The hardware part, which is responsible of capturing the Arabic hand gestures using flex and accelerometer sensors, and Arduino Nano that collect the captured data and send it through Bluetooth to the mobile. The second part is the development of an Android application to process the captured gestures and convert them into texts and auditory voices.

We organized this paper as follows. Section 2 describes the proposed sensory glove and its operation. Section 3 describes, in details, the methodology used in the hardware and software units. Section 4 shows the experimental procedures with the obtained results. Section 5 represents the conclusion of this research.

2. Sensory glove system

Our proposed sensory glove system consists of hardware part and a software part. The hardware part includes the hand gestures acquisition unit, data logger unit, Bluetooth communication unit, and an Android mobile. The software part includes the open-source Arduino Software (IDE), and the cloud-based tool MIT Inventor-2 Software. Fig. 1 shows these two main parts arranged in the transmitter and receiver sides.

The gesture acquisition unit includes the sensory glove and the Arduino Nano that acts as a data logger. We equipped the glove by five Flex sensors and one accelerometer sensor. The Bluetooth communication link is to communicate between the captured gesture (transmitted by *HC-05* module) and the Bluetooth of the Huawei P-smart mobile. We used the MIT Invertor software in developing an Android-mobile Application to receive, process and recognize the *five* Arabic gestures. Then, the application translates the recognized gestures into a displayed texts and a heard sound as well. Fig. 2 shows these details of the proposed system.

3. Methodology

3.1. Selection of Arabic letters

We decided to select five Arabic letters that have unique gestures and orientation to get different readings. Fig. 3 illustrates the selected letters with their gestures and orientations.

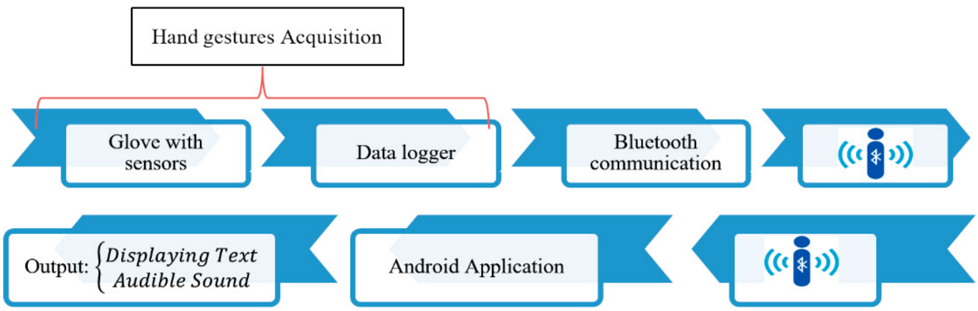

Fig. 1. Main units of the sensory glove system.

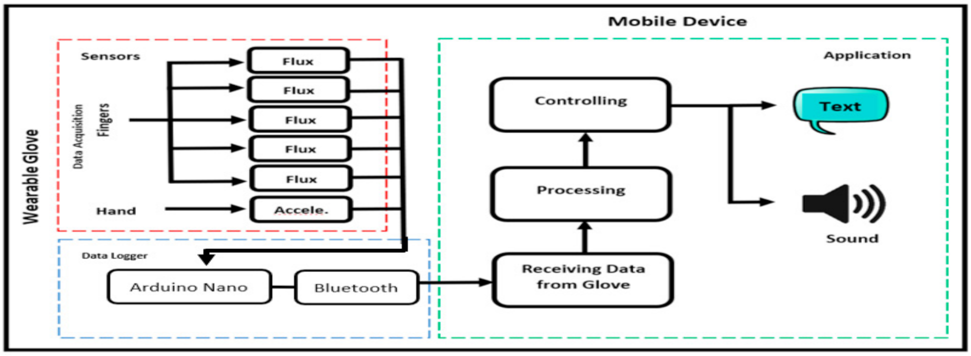

Fig. 2. The proposed system in more details.

3.2. Sensors selections and locations

We used five flex sensors to measure the bindings of each figure and one accelerometer (*ADXL355*) to measure the orientation and the movements of the hand. To make sure that the glove is comfortable and durable, we stitched the flex sensors onto the glove so that user can slide his/her fingers naturally. Similarly, we located the accelerometer at the centre of the glove to avoid and skewness. Fig. 3 displays the locations of these sensors addressed from one to six.

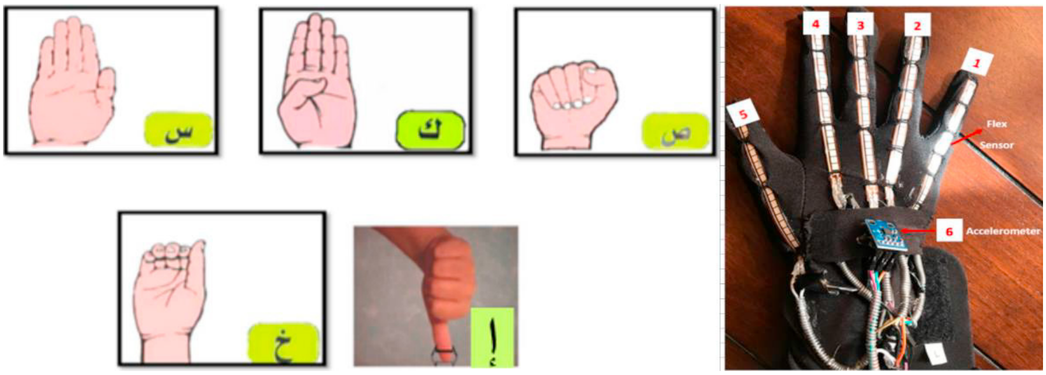

Fig. 3. Selected Arabic letters and the sensors locations on the glove.

3.3. Hardware part

3.3.1. The glove unit

We used a comfortable and easy to carry glove, in Fig 4a.

3.3.2. Sensors unit

We used two types of sensors to detect the gestures and the movements of the hand.

3.3.2.1. Flex sensor

The value of its electrical resistance is a function of its bending. Its size is approximately 0.28 "inch wide and 1"/3"/5" long. Its resistance is up to 125 kΩ with a flat resistance of 25 kΩ with a tolerance of ±30%. It needs an operating voltage of (5 to 12 DCV). It can easily interface with a microcontroller. Fig. 4 shows illustrates the flex sensor when the sensor kept flat and no force applied to it (its resistance is approximately 10 Ω), and when the sensor is bent its resistance between the two output wires is increased proportionally to the amount of bending. The more the bending, the more the value of its resistance [2]. The sensor is in Figs. 4 b and 4c.

3.3.2.2. Accelerometer

Accelerometer is an electromechanical-device measures the acceleration forces. It detects the rotation and motion gestures such as swinging or shaking [3]. Fig. 5 shows the ADXL335-complete 3-axis accelerometer with signal conditioned voltage outputs. Its size is $(4\, mm\, width\, \times\, 4\, mm\, length\, \times\, 1.45\, mm\, hight)$. Its typical current is 350μA with a supply voltage of (1.8 – 5 DCV). The accelerometer is in Fig. 4d.

3.3.3. Bluetooth HC-05

Its module is easily interfaces with Arduino with an operating supply of (1.8- 5 DCV). It has a 2.4 GHz radio transceiver within 10 meters. Its serial port can send up to 25 Mb/s [4]. The Bluetooth module is in Fig. 4e.

3.3.4. Arduino Nano

It is a small-size Arduino with Atmega328 microcontroller. It provides almost the same functions as Arduino Uno. Its operating logic voltage is $(5\, DCV)$and its input voltage is $(7 - 12\, DCV)$. It has *14* digital I/O pins (of which *6* provides PWM output) and *8* analogue input pins. The DC current per I/O pin is 40 mA. Its flash memory is 32 KB and clock speed of 16 MHz [5].The Arduino module is in Figs. 4f and 4g.

3.3.5. Huawei P smart mobile

Its operating system is Android-8 with Octa-core CPU. It has 64 GB memory and 4 $GB\ RAM$. It has $A2DP - 4.2$ Bluetooth. Its screen is 5.65" with a resolution of $(1080\ x\ 2160\ pixels)$, and a 65 dB loud speaker. The mobile is in Fig. 4h.

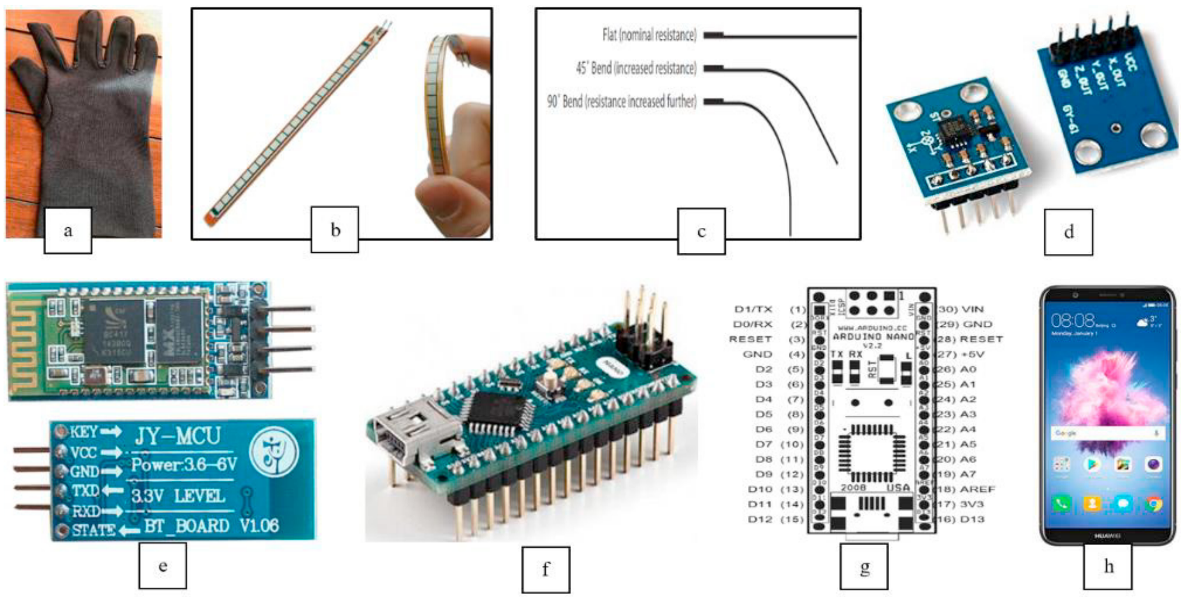

Fig. 4. The hardware components.

3.4. Software part: MIT Inventor

MIT Inventor is a drag-and-drop visual programming tool for designing and building fully functional mobile applications for Android. It promotes a new era of personal mobile computing in which people are empowered to design, create, and use personally meaningful mobile technology solutions for their daily lives, in endlessly unique situations. It allows the developer to focus on the logic for programming an application rather than the syntax of the coding language, fostering digital literacy for all [6].

In our research, we used the MIT invertor to access the features of the mobile such as the Bluetooth, memory, speaker and screen. Within the mobile, the application starts by receiving the gesture data, analyses, recognizes, and sends the corresponding output. Fig. 5 displays the operations of the application, in sequence.

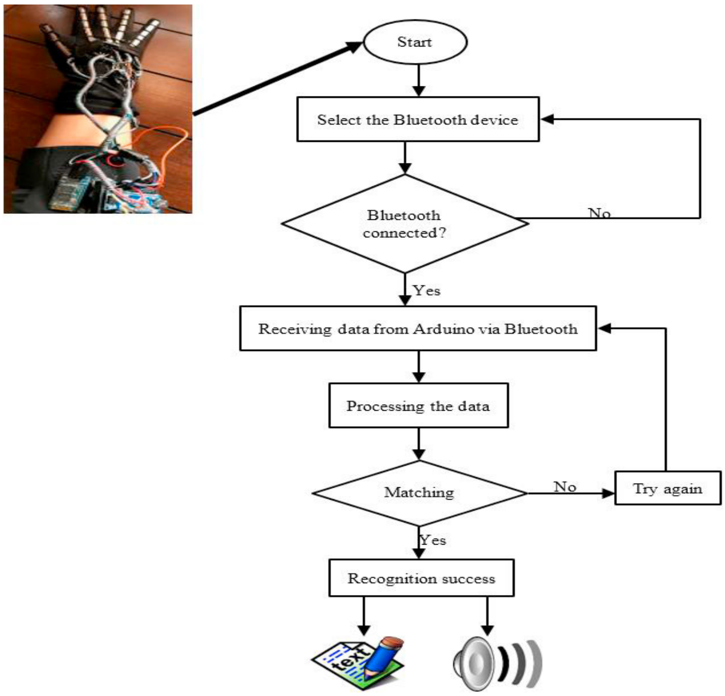

Fig. 5. The software processes in sequence.

4. Procedures and results

4.1. Building dataset

After stitching the six sensors on the glove and for each selected Arabic letter (out of $five$ letters), we recorded the output reading of each sensor by using a voltage divider circuit, equation 1. Then, we calculated the corresponding readings in the digital forms, equation 2. We repeated this task $four$ times by different $four$ subjects and then we calculated the average. Tables $(1 - 6)$ represent the results of the sensors.

$$V_o = V_{CC}\left(\frac{R_2}{R_1 + R_2}\right) \tag{1}$$

$$Digital\ Reading = Analog\,Voltage\ Measured \times \frac{Resolution\,of\ ADC}{System\,Voltage} \tag{2}$$

Table 1. Sensors measurements for the first letter "س".

Sensor Participant	F1 (Little)	F2 (Ring)	F3 (Middle)	F4 (Index)	F5 (Thumb)	F6 (Accelerometer)
Person 1	791	758	799	609	701	200
Person 2	789	759	801	611	702	201
Person 3	786	755	797	608	697	198
Person 4	794	760	800	611	700	201
Average	790	758	800	610	700	200

Table 2. Sensors measurements for the second letter "ك".

Sensor Participant	F1 (Little)	F2 (Ring)	F3 (Middle)	F4 (Index)	F5 (Thumb)	F6 (Accelerometer)
Person 1	791	758	799	609	580	200
Person 2	789	759	801	611	581	201
Person 3	786	755	797	608	578	198
Person 4	794	760	800	611	581	201
Average	790	758	800	610	580	200

Table 3. Sensors measurements for the first letter "ص".

Sensor Participant	F1 (Little)	F2 (Ring)	F3 (Middle)	F4 (Index)	F5 (Thumb)	F6 (Accelerometer)
Person 1	659	600	692	521	580	200
Person 2	660	601	690	520	581	201
Person 3	658	597	687	517	578	198
Person 4	661	602	691	521	581	201
Average	660	600	690	520	580	200

Table 4. Sensors measurements for the first letter "خ".

Sensor Participant	F1 (Little)	F2 (Ring)	F3 (Middle)	F4 (Index)	F5 (Thumb)	F6 (Accelerometer)
Person 1	723	659	738	571	619	200
Person 2	726	660	744	572	622	201
Person 3	722	658	736	566	617	198
Person 4	730	661	742	570	621	201
Average	725	660	740	570	620	200

Table 5. Sensors measurements for the first letter "إ".

Sensor Participant	F1 (Little)	F2 (Ring)	F3 (Middle)	F4 (Index)	F5 (Thumb)	F6 (Accelerometer)
Person 1	659	600	692	521	701	358

Person 2	660	601	690	520	702	360
Person 3	658	597	687	517	697	355
Person 4	661	602	691	521	700	361
Average	660	600	690	520	700	360

Table 6. The average of the five letters.

Letter	F1 (Little)	F2 (Ring)	F3 (Middle)	F4 (Index)	F5 (Thumb)	F6 (Accelerometer)
س	790	758	800	610	700	200
ك	790	758	800	610	580	200
ص	660	600	690	520	580	200
خ	725	660	740	570	620	200
!	660	600	690	520	700	360

4.2. Mapping the Arabic letters and the Bluetooth communication

We used the Arduino Nano module as a data logger between the sensory glove and the Bluetooth module. Fig. 6a shows the interfacing between the sensors and the Arduino Nano. Since the Arduino does not work with Arabic letters, we mapped the five Arabic letters into $five$ English letters represented in table 7 and we programmed the Arduino accordingly. Part of the Arduino code in given in Fig. 6b. Then, the $(HC-05\ )$Bluetooth module sent the mapped data to the mobile application. Fig. 6c shows the interfacing circuit between the Arduino and the Bluetooth module while Fig 7a represents the flowchart of this communication channel. Fig. 7b shows the pairing between the $(HC-05)$module and the mobile Bluetooth.

Table 7. Data mapping.

Letter	Corresponding data
"س"	a
"ك"	b
"ص"	c
"خ"	d
"!"	e

4.3. Software results

We used the MIT invertor to build the welcoming screen, Bluetooth connection, receiving data, recognizing gestures, and sending the output to the screen and the loud speaker. The welcoming screen consists of two buttons, which are the *start* and *exit*, Fig. 8a. We built the Bluetooth connection by both of the "*ListPicker"* and the "*connectivity"*. "*ListPicker"* asks the user to select the Bluetooth device while the "*BluetoothClint1"* from the "*connectivity*" establishes the connection, as shown in Fig. 8b.

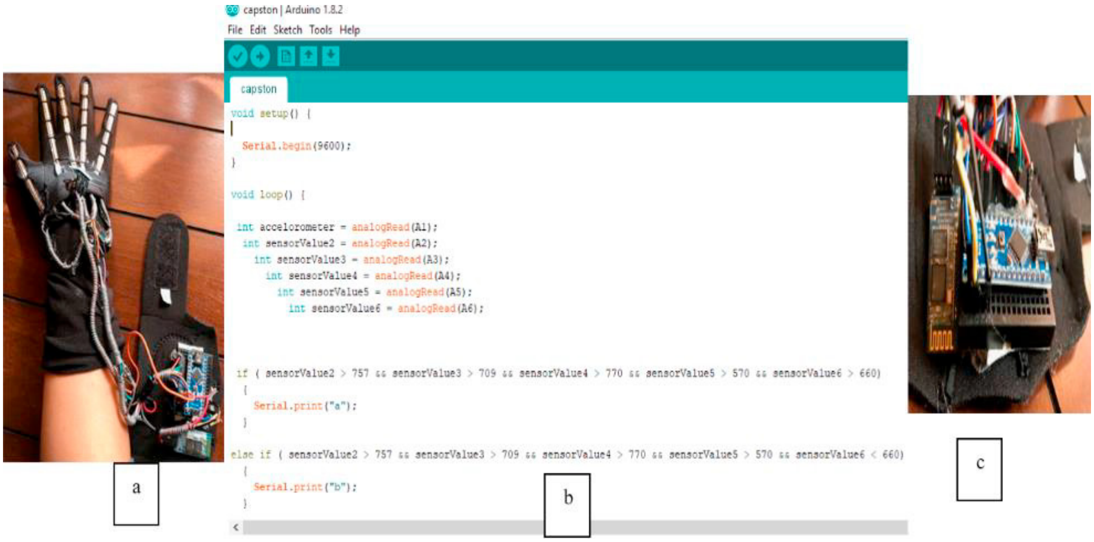

Fig. 6. (a) Interfacing sensors unit with the Arduino. (b) Arduino code. (c). Interfacing Arduino with HC-05 Bluetooth module.

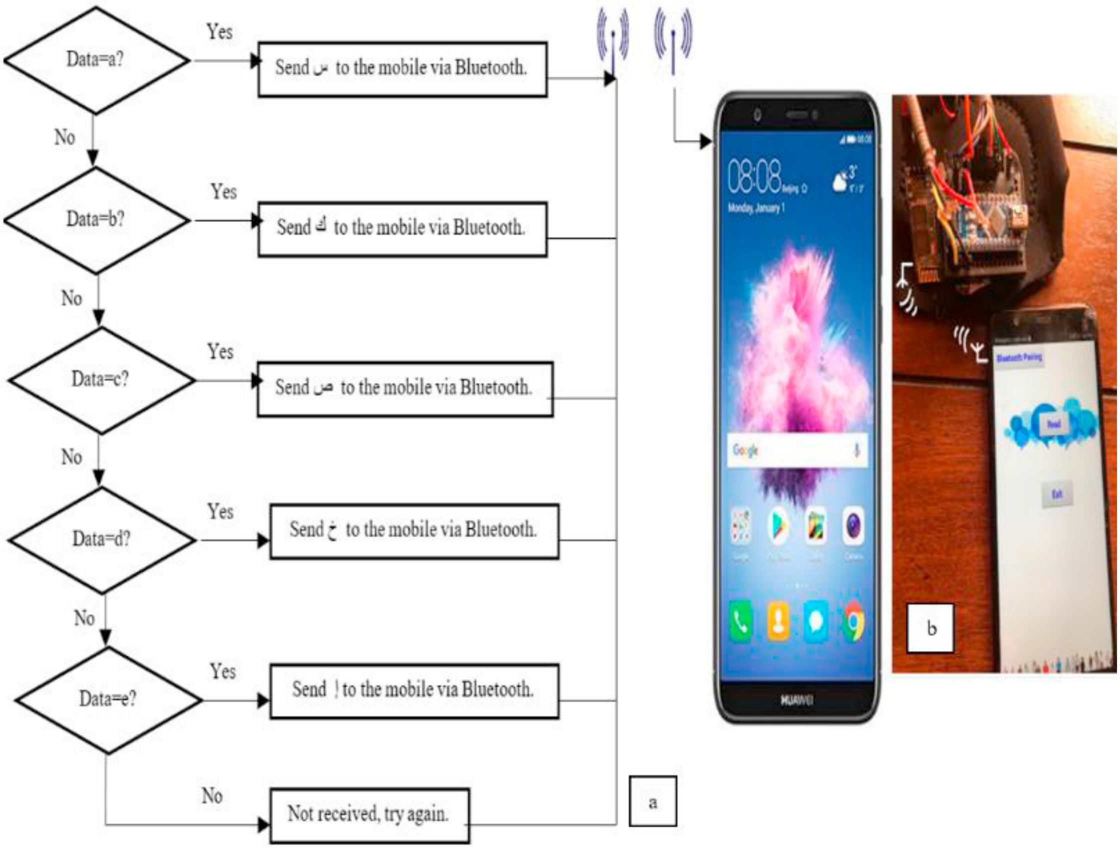

Fig. 7. (a) Flowchart of the Bluetooth communication link. (b). Pairing result between the Bluetooth transceiver.

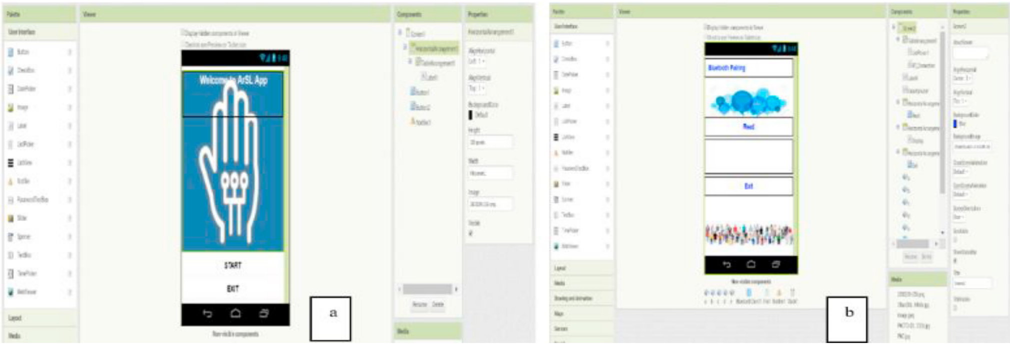

Fig. 8. (a) The welcoming screen. (b) The connection establishment.

Fig. 9a shows the MIT-blocks to list all the available Bluetooth devices. If the user selects one of them, then the connection occurs and the word "*connected*" is displayed; otherwise, "*not connected*" is displayed; Fig. 9b.

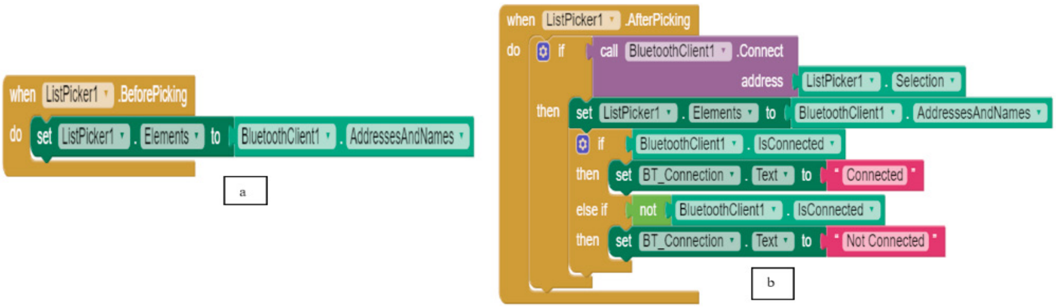

Fig. 9. (a) Displaying all the available devices. (b) User action either to connect or not.

After the establishment of the communication, the application starts receiving the data and the recognition process starts. The recognition process occurs and lastly the generation of the output in the text and audio forms, as in Fig 10a. Fig. 10b shows unrecognized received data from Arduino that is out of the average readings.

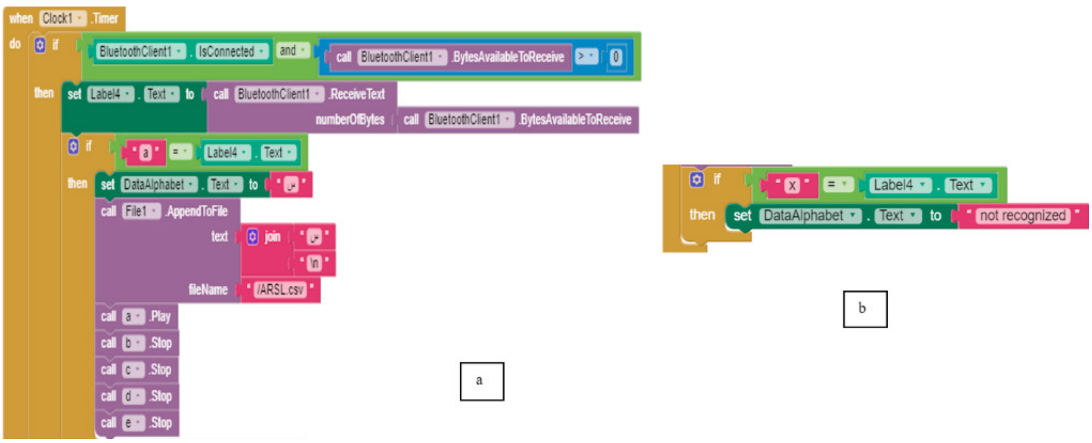

Fig. 10. (a) Recognition process followed by the output generation process. (b). If the acquired data is unrecognized.

Fig. 11a shows the *exit* button asking for the user's confirmation either to close the app or not. If the user confirmed by clicking "*yes*", then the application will be closed. Figs 11b and 11c describe the two buttons used in the welcoming screen, which is the start button to transfer the user to the main screen and exit button to let the user close the app if it pressed by mistake. Fig. 12a shows the whole system while Fig. 12 b shows the result of "س" Arabic letter.

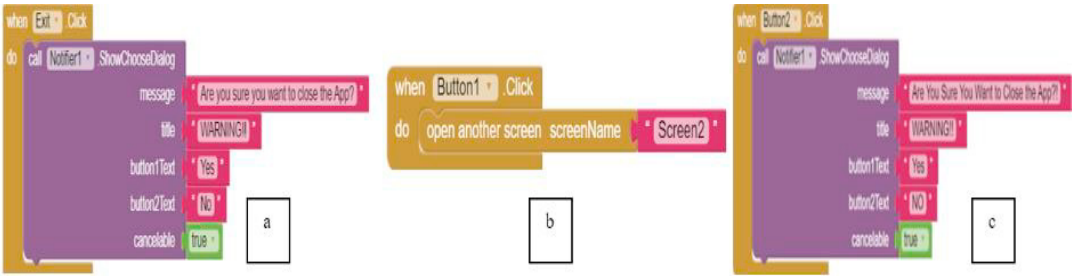

Fig. 11. (a) Exit icon confirmation. (b). Start icon in the welcome screen. (c). Exit icon in the welcome screen.

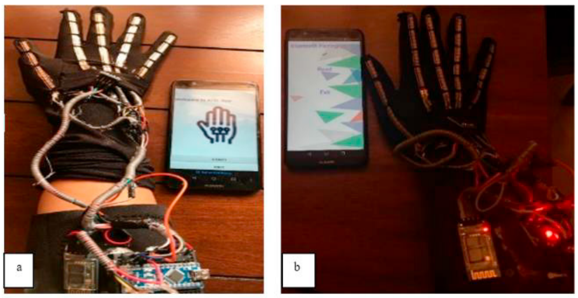

Fig. 12. (a) The whole system. (b) The result of "س" Arabic letter.

5. Conclusion

We designed a sensory glove and developed a mobile application that can help mute people to communicate with normal people in easy way. The glove captures the hand gesture by $five$ flex sensors and one accelerometer and log the acquired data into an Arduino Nano, which interfaces with the $HC - 05$ Bluetooth module. Then, it sends the received data to the mobile application via the Bluetooth link. Inside the mobile, the application receives, recognizes the letter and lastly displays the letter on the screen and activates the speaker as well. Moreover, the user can read the previously recognized signs. We fully succeeded in achieving our goal and the results are optimistic in a way encouraging us to do more improvements on our system. In the future work, we will try implementing the whole list of Arabic letters and the numbers as well. We will try building words not only letters.

References

[1] Deafness Prevention, (2018) "World Health Organization." Available [online]: http://www.who.int/deafness/en/.

[2] Flex Sensor Datasheet, "Flex Sensor." *Spectra Symbol*.

[3] ADXL335 datasheet, (2015) "ADXL335." *Spark Fun*.

[4] Arduino Nano Datasheet, (2012) "Arduino CC." *Arduino Nano*.

[5] HC-05 Datasheet, (2010) "Bluetooth to Serial Port Module-HC-05." *Itead Studio*.

[6] D. Walter and M. Sherman, (2014) "Learning MIT App Inventor: A Hands-On Guide to Building Your Own Android Apps." *Addison-Wesley Professional; 1st Ed.*