International Journal of Innovative Studies in Sciences and Engineering Technology

ISSN 2455-4863 | Open Access | Volume 7, Issue 3, 2021

DOI: https://doi.org/10.20431/2455-4863.070303

Design and Development of Manually Controlled Swab Collecting Robot

G.S.Nirmala[1], S.B.Jamadade[1], R.S.Bhosale[1], M.S.Patil[1], S.U.Mohite[1], A.A.Jadhav[2]

[1]*Student, Mechanical Engineering, ADCET, Ashta, Maharashtra, India*

[2]*Assistant Professor, Mechanical Engineering, ADCET, Ashta, Maharashtra, India*

ABSTRACT

Nasopharyngeal (NP) swab sampling is an effective approach for the diagnosis of coronavirus disease 2019 (COVID-19). Medical staffs carrying out the task of collecting NP specimens are in close contact with the suspected patient, thereby posing a high risk of cross-infection. We propose a low-cost miniature robot that can be easily assembled and remotely controlled. The system includes an active end-effector, a passive positioning arm, and a detachable swab gripper with integrated force sensing capability. Testing the coronavirus patient the swab is collected first. While collecting the swab of patients many medical health workers get infected due to those infected persons. This will increase the stress on medical healthcare. The frontline workers and the other doctors and their companions are mostly vulnerable to this situation. To deal with this problem we come up with the solution of swab collecting robot which can be operated manually using Bluetooth and mobile application without the vicinity of the any person being in the presence during test. This is the main objective of the project that is eliminating the person from the testing area so that none of the healthcare worker get infected while collecting the swab.

Keywords: Robot,swab,healthcare,doctors.

INTRODUCTION

The global outbreak of novel coronavirus pneumonia (NCP) caused by coronavirus disease 2019 (COVID-19) has spread rapidly. Collection of specimens from the surface of the respiratory mucosa with nasopharyngeal (NP) or Oropharyngeal (OP) swabs are treated as effective ways for the diagnosis and screening [1]. Several recent studies have

indicated that OP swabs are less effective than NP swabs in detecting the COVID-19 virus [2, 3] and concluded that the

use of NP may be more suitable, although a study also highlighted the data should be viewed with cautions [4]. According to US CDC, both NP and OP should be performed by a healthcare professional. Other possible approaches, e.g., nasal mid-turbinate (NMT) swab (also known as deep nasal swab) and anterior nares (nasal swab) specimen, could be supervised onsite self-collection and home self-collection [5]. During the conventional manually controlled swab sampling, medical staffs are inevitably in close contact with the suspected patient, posing a high risk of cross-infection. The operating skills and psychological states of medical workers may also affect the accuracy and quality of the results of swab collection.

LITERATURE REVIEW

Dr.P.Gomathietel [1] describes a mechanized arm with four degree of freedom is formed and can pick the things with a specific weight and spot them in a looked-for region. To empower the lifting of the things, Servomotors is used. The composing PC programs is done on ATMEGA-328 Microcontroller using Arduino programming. The Potentiometers recognize the edge of the unrest and the signs are shipped off the Microcontroller in like way. In the domain of apply self-rule, this Robotic arm has been wound up being smart. This kind of the arms have various applications in the field of mechanical apply self-governance where the robotization is required. Mechanized pick and spot a structure contain a stacking station, testing station, planning station, and arranging station.Swarna Prabha Jena et el. [2] This paper depicts about the motion control robot which can be constrained by your ordinary hand motion. It comprises of fundamentally two sections, one is transmitter part and another is collector part. The transmitter will communicate the sign as indicated by the situation of accelerometer and your hand motion and the beneficiary will get the sign and make the robot move individual way. Here, the program is planned by utilizing Arduino IDE. Harish K et el .[3] The paper depicts proposed idea of pick and spot robot utilizing Arduino is executed through RF play station. It is tracked down that, the robot so carried out can find itself to the area where the item to be lifted is accessible with the assistance of frame and four dc motors. Further contingent on controlling activity gave to servo motor it lifts the item and finds something similar at required objective. Ashly Baby1 et el.[4] describes A mechanical arm is carried out utilizing Arduino to pick and place objects the more securely without causing a lot of harm. The automated arm utilized here contain a delicate getting gripper which securely handle the object. By the utilization of this item the modern activities or complex tasks and risky activities should be possible effectively and securely in a limited ability to focus time. The utilization of delicate getting gripper and low force remote correspondence procedure like Bluetooth makes framework more successful when contrasted with different frameworks. The proposed framework is equipped for lifting just little loads, by presenting high force giving motor huge loads can be picked. The range is additionally an impediment it very well may be upgraded by utilizing a remote-control technology.Reza Ezuan Saminb et el.[5] Published by Elsevier Ltd. It gives Generally, this undertaking can be partitioned into two significant areas that are equipment improvement and programming advancement. The equipment activities incorporate the robotization interaction of controlling servo motor and further more foster the mechanical arm connection and links. Programming improvement comprises of fostering the web worker and furthermore programming the Arduino Uno. From the investigations that have been made, it's unmistakably shows that controlling a servo motor is simple and the yield is precise. Accordingly, it is the correct decision to pick servo motor for the actuator of the robot arm. The reason for this undertaking is to show that robots confined to modern utilization just as well as appropriate for family use. Exploiting the far and wide utilization of web availability these days, robots can be controlled by means of web rather than a devoted regulator only for the robots.

ARDUINO PROGRAM

Arduino is an open-source prototyping stage considering easy to use hardware and programming. Arduino sheets can peruse. Data sources lighting on a sensor, a finger on a catch, or a twitter message and change it into a yield starting a motor, turning on a LED , circulated something on the web.

```
#include <SoftwareSerial.h>
#include <Servo.h>
Servo s01;
Servo s02;
Servo s03;
Servo s04;
Servo s05;
Servo s06;
Software Serial Bluetooth(3, 4); // Arduino(RX, TX) - HC-05 Bluetooth (TX, RX)
int servo1Pos, servo2Pos, servo3Pos, servo4Pos, servo5Pos, servo6Pos; // current position
int servo1PPos, servo2PPos, servo3PPos, servo4PPos, servo5PPos, servo6PPos; // preposition
int s01SP[50], s02SP[50], s03SP[50], s04SP[50], s05SP[50], s06SP[50]; // for storing positions
int speedDelay = 20;
int index = 0;
String dataIn = "";
void setup() {
s01.attach(5);
s02.attach(6);
s03.attach(7);
s04.attach(8);
s05.attach(9);
```

```
s06.attach(10);
Bluetooth.begin(38400); // Default baud rate of the Bluetooth module
Bluetooth.setTimeout(1);
delay(20);
//  arm initial position
servo1PPos = 90;
servo01.write(servo1PPos);
servo2PPos = 150;
servo02.write(servo2PPos);
servo3PPos = 35;
servo03.write(servo3PPos);
servo4PPos = 140;
servo04.write(servo4PPos);
servo5PPos = 85;
servo05.write(servo5PPos);
servo6PPos = 80;
servo06.write(servo6PPos);
}
void loop() {
if (Bluetooth.available() > 0) {
dataIn = Bluetooth.readString(); // Read the data as string
// If "Waist" slider has changed value - Move Servo 1 to position
if (dataIn.startsWith("s1")) {
String dataInS = dataIn.substring(2, dataIn.length()); // Extract only the number. E.g. from "s1120" to "120"
servo1Pos = dataInS.toInt(); // Convert the string into integer
// We use for loops so we can control the speed of the servo
// If previous position is bigger than current position
if (servo1PPos > servo1Pos) {
for ( int j = servo1PPos; j >= servo1Pos; j--) { // Run servo down s01.write(j);
delay(20); // defines the speed at which the servo rotates
}
}
// If previous position is smaller then current position
if (servo1PPos < servo1Pos) {
for ( int j = servo1PPos; j <= servo1Pos; j++) { // Run servo up
s01.write(j);
```

```
delay(20);
}
}
servo1PPos = servo1Pos; // set current position as preposition
}
// Move Servo 2
if (dataIn.startsWith("s2")) {
String dataInS = dataIn.substring(2, dataIn.length());
servo2Pos = dataInS.toInt();
if (servo2PPos > servo2Pos) {
for ( int j = servo2PPos; j >= servo2Pos; j--) {
s02.write(j);
delay(50);
}
}
if (servo2PPos < servo2Pos) {
for ( int j = servo2PPos; j <= servo2Pos; j++) {
s02.write(j);
delay(50);
}
}
servo2PPos = servo2Pos;
}
// Move Servo 3
if (dataIn.startsWith("s3")) {
String dataInS = dataIn.substring(2, dataIn.length());
servo3Pos = dataInS.toInt();
if (servo3PPos > servo3Pos) {
for ( int j = servo3PPos; j >= servo3Pos; j--) {
s03.write(j);
delay(30);
}
}
if (servo3PPos < servo3Pos) {
for ( int j = servo3PPos; j <= servo3Pos; j++) {
s03.write(j);
```

```
delay(30);
}
}
servo3PPos = servo3Pos;
}
// Move Servo 4
if (dataIn.startsWith("s4")) {
String dataInS = dataIn.substring(2, dataIn.length());
servo4Pos = dataInS.toInt();
if (servo4PPos > servo4Pos) {
for ( int j = servo4PPos; j >= servo4Pos; j--) {
s04.write(j);
delay(30);
}
}
if (servo4PPos < servo4Pos) {
for ( int j = servo4PPos; j <= servo4Pos; j++) {
s04.write(j);
delay(30);
}
}
servo4PPos = servo4Pos;
}
// Move Servo 5
if (dataIn.startsWith("s5")) {
String dataInS = dataIn.substring(2, dataIn.length());
servo5Pos = dataInS.toInt();
if (servo5PPos > servo5Pos) {
for ( int j = servo5PPos; j >= servo5Pos; j--) {
s05.write(j);
delay(30);
}
}
if (servo5PPos < servo5Pos) {
for ( int j = servo5PPos; j <= servo5Pos; j++) {
s05.write(j);
```

```
delay(30);
}
}
servo5PPos = servo5Pos;
}
// Move Servo 6
if (dataIn.startsWith("s6")) {
String dataInS = dataIn.substring(2, dataIn.length());
servo6Pos = dataInS.toInt();
if (servo6PPos > servo6Pos) {
for ( int j = servo6PPos; j >= servo6Pos; j--) {
s06.write(j);
delay(30);
}
}
if (servo6PPos < servo6Pos) {
for ( int j = servo6PPos; j <= servo6Pos; j++) {
s06.write(j);
delay(30);
}
}
servo6PPos = servo6Pos;
}
// If button "SAVE" is pressed
if (dataIn.startsWith("SAVE")) {
s01SP[index] = servo1PPos; // save position into the array
s02SP[index] = servo2PPos;
s03SP[index] = servo3PPos;
s04SP[index] = servo4PPos;
s05SP[index] = servo5PPos;
s06SP[index] = servo6PPos;
index++; // Increase the array index
}
// If button "RUN" is pressed
if (dataIn.startsWith("RUN")) {
runservo(); // Automatic mode - run the saved steps
```

```
}
// If button "RESET" is pressed
if ( dataIn == "RESET") {
memset(s01SP, 0, sizeof(s01SP)); // Clear the array data to 0
memset(s02SP, 0, sizeof(s02SP));
memset(s03SP, 0, sizeof(s03SP));
memset(s04SP, 0, sizeof(s04SP));
memset(s05SP, 0, sizeof(s05SP));
memset(s06SP, 0, sizeof(s06SP));
index = 0; // Index to 0
}
}
}
// Automatic mode custom function - run the saved steps
void runservo()
{
while (dataIn != "RESET") { // Run the steps repetadly again until "RESET" button is pressed
for (int i = 0; i <= index - 2; i++) { // Run through all steps(index)
if (Bluetooth.available() > 0) { // Check for incomding data
dataIn = Bluetooth.readString();
if ( dataIn == "PAUSE") { // If button "PAUSE" is pressed
while (dataIn != "RUN") { // Wait until "RUN" is pressed again
if (Bluetooth.available() > 0) {
dataIn = Bluetooth.readString();
if ( dataIn == "RESET") {
break;
}
}
}
}
// If speed slider is changed
if (dataIn.startsWith("ss")) {
String dataInS = dataIn.substring(2, dataIn.length());
speedDelay = dataInS.toInt(); // Change servo speed (delay time)
}
}
```

```
// Servo 1
if (s01SP[i] == s01SP[i + 1]) {
}
if (s01SP[i] > s01SP[i + 1]) {
for ( int j = s01SP[i]; j >= s01SP[i + 1]; j--) {
s01.write(j);
delay(speedDelay);
}
}
if (s01SP[i] < s01SP[i + 1]) {
for ( int j = s01SP[i]; j <= s01SP[i + 1]; j++) {
s01.write(j);
delay(speedDelay);
}
}
// Servo 2
if (s02SP[i] == s02SP[i + 1]) {
}
if (s02SP[i] > s02SP[i + 1]) {
for ( int j = s02SP[i]; j >= s02SP[i + 1]; j--) {
s02.write(j);
delay(speedDelay);
}
}
if (s02SP[i] <  s02SP[i + 1]) {
for ( int j = s02SP[i]; j <= s02SP[i + 1]; j++) {
s02.write(j);
delay(speedDelay);
}
}
// Servo 3
if (s03SP[i] == s03SP[i + 1]) {
}
if (s03SP[i] > s03SP[i + 1]) {
for ( int j = s03SP[i]; j >= s03SP[i + 1]; j--) {
s03.write(j);
```

```
delay(speedDelay);
}
}
if (s03SP[i] < s03SP[i + 1]) {
for ( int j = s03SP[i]; j <= s03SP[i + 1]; j++) {
s03.write(j);
delay(speedDelay);
}
}
// Servo 4
if (s04SP[i] == s04SP[i + 1]) {
}
if (s04SP[i] > s04SP[i + 1]) {
for ( int j = s04SP[i]; j >= s04SP[i + 1]; j--) {
s04.write(j);
delay(speedDelay);
}
}
if (s04SP[i] < s04SP[i + 1]) {
for ( int j = s04SP[i]; j <= s04SP[i + 1]; j++) {
s04.write(j);
delay(speedDelay);
}
}
// Servo 5
if (s05SP[i] == s05SP[i + 1]) {
}
if (s05SP[i] > s05SP[i + 1]) {
for ( int j = s05SP[i]; j >= s05SP[i + 1]; j--) {
s05.write(j);
delay(speedDelay);
}
}
if (s05SP[i] < s05SP[i + 1]) {
for ( int j = s05SP[i]; j <= s05SP[i + 1]; j++) {
s05.write(j);
```

```
delay (speed Delay);
}
}
// Servo 6
if (s6SP[i] == s06SP[i + 1]) {
}
if (s06SP[i] > s06SP[i + 1]) {
for ( int j = s06SP[i]; j >= s06SP[i + 1]; j--) {
s06.write(j);
delay(speedDelay);
}
}
if (s06SP[i] < s06SP[i + 1]) {
for ( int j = s06SP[i]; j <= s06SP[i + 1]; j++) {
s06.write(j);
delay(speedDelay);
}
}
}
}
```

CONCLUSION

From above work it can be concluded that, the swab collecting robot can satisfactorily and effectively operated to collect the throat swab of covid-19 patient. The efficiency and effectiveness of a robot depends on the operator which is going to operate the robot via mobile application. The robot can be operated within a range of 10 meter as the Bluetooth module range is 10 meter. Direct power supply can we keep one from the adaptor to the robot which make it feasible to use at any location where electricity is available. From this the main objective is accomplished that is collecting throat swab without being the medical healthcare workers in a vicinity of covid-19 patient. If it is used effectively swab can be collected within the 1 min time period with no harm to patient.

REFERENCES

1. Design and Implementation of Pick and Place Robot Using Arduino for Smart Grid Monitoring International Journal of Advanced Information Science andTechnology"Dr.P.Gomathi, 2S.Baskar (IJAIST) ISSN: 2319:2682 Vol.6, No.9, September 2017 DOI:10.15693

2. Computer assisted robotic examination swab sampling" Gerhart, Grant R.; Gage, Douglas W. (CARESS) Publication:Unmanned Systems Technology X. Edited by ; Shoemaker, Charles M. Proceedings of the SPIE, Volume 6962, article id. 69620O,9pp.(2008),PubnDate:April2008,DOI:10.1117/12.783229 Bibcode:2008SPIE.6962E..0OH

3. Intelligent robots for throat swab sampling of coronavirus tests," Q. Wang,Global Times, Mar. 2020.

4. Laboratory testing for coronavirus disease 2019 (COVID-19) in suspected human cases: interim guidance, World Health Organization (No. WHO/COVID-19/laboratory/2020.4), Mar. 2020

5. Pick and Place Robotic Arm Using Arduino." Harish K, Megha D, Shuklambari M, Amit K, Chaitanya K Jambotkar. International Journal of Science, Engineering and Technology Research (IJSETR)Volume 6, Issue 12, December

2017,

6. Combating COVID-19—The role of robotics in managing public health and infectious diseases," G. Yangetal., Sci. Robot., vol. 5, no. 40, eabb5589, Mar. 2020. DOI: 10.1126/scirobotics.abb5589

7. Design of a Low-cost Miniature Robot to Assistthe COVID-19 Nasopharyngeal Swab Sampling"Shuangyi Wang, Kehao Wang, Hongbin Liu and Zengguang Hou,IEEE Transactions on Medical Robotics and Bionics,

Citation: *G.S.Nirmala, S.B.Jamadade, R.S.Bhosale, et.al. Design and Development of Manually Controlled Swab Collecting Robot. Int J of Innovative Studies in Sciences and Engineering Technology. 2021; 7(3): 13-23. DOI: https://doi.org/10.20431/2455-4863.070303.*