[Hand holding simulator] - [Embracing social distancing]

David Östling, Isak Sundström (Group 23)
{dostl@kth.se, isaksun@kth.se}

ABSTRACT

The two groups worked together to create a "*hand holding simulator*". It worked by the use of a time of flight- as well as a presence sensor. When a hand was placed close enough to one of the sensors, it triggered a corresponding servo on the other side of a radio signal to go off. Each servo then placed an artificial hand on each user's other hand to make it feel like the users were touching. To add upon this a joystick and a button were used to trigger buzzers on each side of a radio signal to alert the each user that they want to hold hands.

Initial idea and design process

The main source of motivation regarding this project is the currently active coronavirus pandemic. Both groups agreed that we wanted to create something to make this mess seem a little bit more bright and enjoyable. The groups' initial idea was to create a *"hand petting machine"*. This would function by one group requesting (e.g. with a button press) a "pet" from the other group and if they were to accept the request (by e.g. pulling a joystick) a servo would trigger. As a result an artificial hand would land on the user on the requesting ends' shoulder. The idea was to create a playful experience of social interaction. The group figured this would greatly suit the current virus situation - basically a wacky way of embracing social distancing.

Another idea was to use the same sort of request mechanism but instead of using the artificial hand, the group could use a matrix screen. The idea was to display some sort of message (a heart or similar) when the request was accepted and maybe even put the screen on a bracelet or alike. But the group soon realized the drawbacks of these displays, they were very complex and required far too many pins to function properly as well as an even more complex code to go with it. This resulted in the group scrapping this plan.

We decided on developing the first idea with some modifications. We would use a presence sensor on one end and a distance sensor on the other to control servos that would make a artificial hand come down and hold the other person's hand, creating an intimate interaction from a distance (see figure 2 for the action). We also wanted to use buzzers to make some sort of ringtone to alert the other person that it was time to hold hands/link up. Our group first wanted to use a sound sensor as the input for this ringtone to send to the other person, but it proved very challenging to use both the sound sensor and the distance sensor in one arduino. Therefore, we unfortunately had to settle with using a button instead.

Implementation and design

Our group's final version of the project was built using one time of flight sensor, one radio transceiver, one buzzer, one button and a micro servo. The button was used to send a ringtone over to the other group's buzzer to alert them that our side wants to hold hands. Their side functioned similarly but instead of using a button they used a joystick to create a modified ringtone dependant on the X values of the joystick. After receiving a call each group had to act in order for the project to work as intended. The time of flight sensor on our side (as well as the presence sensor on the other group's side) was used to check whether or not a hand was in range of the little heart on the box. The input from the sensors would be sent to the other group with radio transceivers, both ways simultaneously, to control each other's' servos and buzzers.

Our distance sensor measures distance in millimeters. We capped it at 180mm so that it would only send signals when the hand was close, and so that the other group's servo wouldn't rotate too much when a hand was not near. We also added a simple low pass filter to our distance sensor values in order to make it a little bit smoother. With our button we simply sent a 1 when the button was not pressed and a 0 when it was pressed. From the other group we got values from their presence sensor and their joystick. In neutral mode (when not interacting), the joystick sent us values around 500. We added thresholds at 450 and 550 and put those values straight into our buzzer using the tone() function, so that when the other group moved the joystick outside the thresholds, our buzzer would make sounds. We also put a low pass filter on the values from their presence sensor, to make it more smooth when feeding into our servo. Also, because the presence sensor values were much too high to feed directly into our servo, we divided them by 30, which put the highest values just over 180 (which is the highest angle for the servo.). Our servo then had a range from 90 to 180 degrees depending where the other group had their hand. In order to use the transceiver bilaterally (sending and receiving data from both direction at the same time) we took help from this[1] tutorial.

Because of the limited number of wires, we had to put the button and buzzer inside the box with the arduino and breadboard (see Figure 4), we used a lever that you could flip outside the box in order to reach the button.

---

1 https://howtomechatronics.com/tutorials/arduino/arduino-wireless-communication-nrf24l01-tutorial/ - radio transciever

Figure 1: “The setup!”

Figure 2: “Both systems working together - holding hands”

Figure 3: “United! (the other group is making the yellow hand come down, and the left hand in the picture is doing the same for the other group.)”

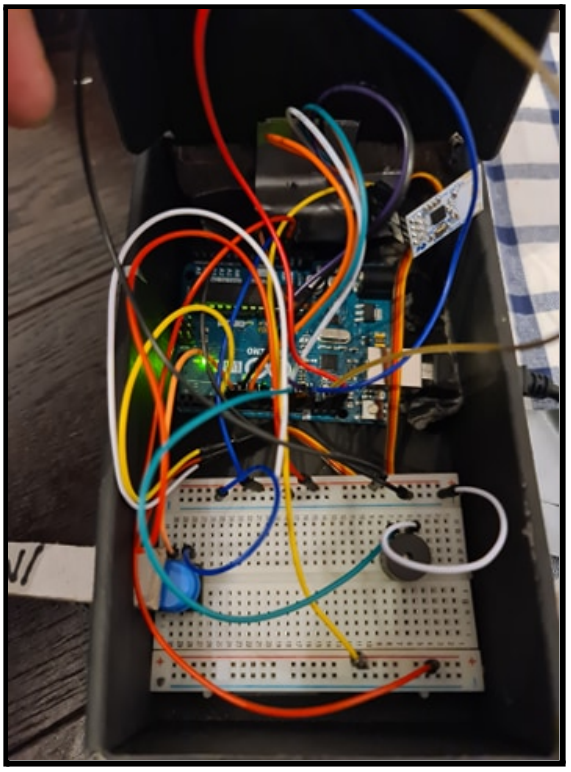

Figure 4: “The wiring”

Sensors used in our design

VL53L0X - Time of flight[2]

Uses a small laser to track the distance between the sensor and an object in range.

nRF24L01- Radio transceiver

Uses radio waves to connect to another transceiver. It allows bilateral communication if used correctly.

SG90 - Microservo[3]

A small rotary device/actuator that allows for control of angular position.

Actuators used

Buzzer[4]

A small actuator that produces sounds in different frequencies depending on the given a value (Hz).

Button[5]

A small actuator triggered by a push/press. When pressed it yields the value 1, and when not pressed it retains the value 0.

Provided UX and Motivation

The experience provided by our design coupled with the other groups design is a way of sharing affection by touch from a distance. The coronavirus pandemic has forced many people to stay alone inside their apartments and avoid all social contact. Now, one can already talk on the phone or on skype in order to achieve at least some amount of social contact, but with our design one can also touch each other's hands in a way. On our end, you put your hand on a designated area and when the person using the other group's design puts their hand on a sensor, a little hand on a stick will come down and touch our hand *(see Figure 3)*. It also works almost identically on the other end. When we come near a distance sensor on our end, another hand on a stick will descend on the other group's hand. When this is done simultaneously on both sides, it will provide an experience of touching each other. We also have a switch that will send a little ringtone to the other group, signaling that we want to hold their hand. Vice versa, we also have a buzzer that the other group can send data to in order to call on us.

In our opinion this is a small but sweet little interaction that provides an enjoyable and intimate experience, especially for someone who can not leave their home or have visitors.

Group Member Contributions

Since the majority (3/4) of the two groups were stationed in Stockholm it was easy to meet up in real life to discuss ideas, sensors and wiring. Our two groups worked very closely to each other in this project, since there is practically no functionality in the design unless both systems are used at the same time. David, Isak and Johan performed the wiring and coding together whereas Adam (who is currently stationed in Örebro) contributed with project ideas and code through Skype. The group took turns wiring, documenting and coding so that everyone felt equally included in the project. In group 23 (our group) the majority of the coding was done by Isak while David stood for the wiring and sensor research.

Reflections and learnings

Both groups agree to have learned a lot more about arduino, both coding and wiring. When it comes to the relation to the course intended learning outcomes we believe to have met all expectations. We believe this was achieved by a lot by trial and error regarding all given hardware. Both groups tested a lot of different sensors to perform the same task and we believe we singled out the peak performers for each function. The project proved to be a major test of our arduino knowledge acquired from the laborations. Putting these skills to the test proved very challenging but also very rewarding.

While being quite fragile at times, cardboard proved to be a very usable material. We learned that cardboard is very manageable even with entry level tools and allowed the groups to create the design of our project without too much hassle. Any mistakes made with the cardboard cutouts were also quite easy to deal with (using tape).

The other main material used was cloth from old pillowcases and such. While cardboard might be a very good material for shaping it can certainly be an eyesore. We used the colorful cloth to brighten up our build somewhat. We wanted our project to emit a feeling of happiness for the user, not a boring cardboard feeling. We used different patterns of cloth to form the foundation for our build while also making user guidelines (for example a drawn hand symbol where the hand is supposed to be placed).

Both groups are satisfied with the materials used especially since they are taken from used products (shoe boxes, old pillowcases and similar). However we believe the build could benefit from some more durable materials like wood. This would enable safer transportation of the project without risking any damage to the wiring or the cardboard holding

---

2 https://learn.adafruit.com/adafruit-vl53l0x-micro-lidar-distance-sensor-breakout/arduino-code - time of flight sensor tutorial

3 https://www.instructables.com/id/How-to-usa-a-Servo-motor-Arduino-Tutorial/ - servo

4 https://create.arduino.cc/projecthub/SURYATEJA/use-a-buzzer-module-piezo-speaker-using-arduino-uno-89df45 - buzzer tutorial

5 https://arduinogetstarted.com/tutorials/arduino-button-led - button tutorial

everything together. This would require a stronger servo that could hold the weight.

Both groups were somewhat aggravated by a couple of factors. The first one being to make all sensors talk with each other simultaneously though the transceivers without crashing the program/arduino. It was very hard to achieve a stable connection between all sensors at times and even worse - to add new sensors to the equation. Sending data from multiple sensors at once proved to be very challenging. This consumed a lot of our time and caused a lot of errors in the code but in the end both groups are pleased with the final result.

For better user experience the lever for the button could be improved upon. It works well as it is but a more traditional button layout could prove to be more user friendly rather than pulling a lever upwards to trigger the action. Better looking artificial hands would also improve the experience quite a bit. Our little cardboard hand do a somewhat poor job on selling the illusion.

One thing that could greatly improve the “hand holding simulator” would be to use a radio transceiver with better signal range. This way people could be sitting further away from each other while using the simulator. The current sensors are limited to 100 meters[6] which would suit them for use in the same living space or between close neighbours. However we believe our project could have substantially benefited from the use of a WIFI module/sensor[7]. As a result both arduino kits have to be connected to a WIFI router in order to work properly. This means that our hand holding simulator is dependant on a stable WIFI connection at all times but this way range is no longer an issue. We believe that in today's society this is a reasonable priority measure.

The old fashioned button tends to be a good choice in a lot of situations. However we believe that if we would have had more time, the use of a sound sensor absolutely could have improved the “*ringtone*” making it feel a little more personal.. The button limits the ringtone to a single tone- or set of tones. If a sound sensor[8] would have been implemented it could function as an e.g. artificial/digital flute. This could be used to form a series of notes coming directly from the user making sounds into the sensor. Adding this to the project would have probably enlightened the user experience a whole lot!

For some reason that neither group has been able to figure out, the two arduinos sometimes stop talking to each other for no apparent reason. Something like that could possibly be solved in the code, how to do it is however uncertain. One hypothesis we have is that there might have been a backlash/short somewhere in the wiring connected to the radio transceivers. Also, the servo sometimes felt a little bit choppy in its movement which could maybe be solved with a better filter than our simple low pass.

A final thought was also to put the pressure- and distance sensors where the user were supposed to lay their hands. This would allow one hand usage but we figured this would cause damage to our micro servo since we would have to force it up the first bit in order for the sensor to read the appropriate values and move on its own. This could probably have been tinkered with in the code but we figured the two handed use was more suited for the current build. However we believe a solution to this could have been to use a more durable and stronger servo. This would have also allowed for a more human like hand (design purpose), since the current servo struggled with anything heavier than cloth and cardboard.

REFERENCES

1. https://learn.adafruit.com/adafruit-vl53l0x-micro-lidar-distance-sensor-breakout/arduino-code - time of flight sensor tutorial
2. https://www.instructables.com/id/How-to-usa-a-Servo-motor-Arduino-Tutorial/ - servo
3. https://create.arduino.cc/projecthub/SURYATEJA/use-a-buzzer-module-piezo-speaker-using-arduino-uno-89df45 - buzzer tutorial
4. https://arduinogetstarted.com/tutorials/arduino-button-led - button tutorial
5. https://howtomechatronics.com/tutorials/arduino/arduino-wireless-communication-nrf24l01-tutorial/ - radio transciever
6. https://howtomechatronics.com/tutorials/arduino/arduino-wireless-communication-nrf24l01-tutorial/ - provides radio signal range limit
7. https://www.kjell.com/se/produkter/el-verktyg/arduino/moduler/wifi-modul-for-arduino-esp8266-p87947 - WIFI module example
8. https://www.kjell.com/se/produkter/el-verktyg/arduino/tillbehor/ljudsensormodul-for-arduino-p87071 - Sound sensor example

---

6 https://howtomechatronics.com/tutorials/arduino/arduino-wireless-communication-nrf24l01-tutorial/ - provides radio signal range limit

7 https://www.kjell.com/se/produkter/el-verktyg/arduino/moduler/wifi-modul-for-arduino-esp8266-p87947 - WIFI module example

8 https://www.kjell.com/se/produkter/el-verktyg/arduino/tillbehor/ljudsensormodul-for-arduino-p87071 - Sound sensor example