

Computer Vision System for a wheelchair for children with functional diversity

Jaime Cabrera Calderón

Abstract–

This paper contains all the information related with the definition, development and results of a Computer Vision System project that will be implemented in an existing wheelchair. The project is based on a low cost philosophy as is the existing wheelchair as well. The main objective of the project is to provide the wheelchair with functionalities that allow giving a certain autonomy to children who do not have any, being a project aimed at solving a real need of the community and improving the living conditions of children with functional diversity. Children can choose which corner of the room to go to, where there will be a visual stimulus, and the wheelchair goes to that place.

Keywords–

Low-cost, Computer Vision, QR code, Bluetooth, Localization

Resum–

Aquest document conté tota la informació relacionada amb la definició, el desenvolupament i els resultats d'un projecte de sistema de visió artificial que s'implementarà en una cadira de rodes existent. El projecte es basa en una filosofia de baix cost, com també ho és la cadira de rodes existent. L'objectiu principal del projecte és proporcionar a la cadira funcionalitats que permetin atorgar certa autonomia als nens que no tenen cap, sent un projecte destinat a resoldre una necessitat real de la comunitat i millorar les condicions de vida dels nens amb pluridiscapacitat. Els nens poden triar a que cantonada de l'habitació anar, on hi haurà un estímul visual, y la cadira de rodes anirà al lloc.

Paraules clau–

Low-cost, Computer Vision, QR code, Bluetooth, Localization



-
- E-mail de contacte: jaime.cabrera@e-campus.uab.cat
 - Menció realitzada: Enginyeria de Computació
 - Treball tutoritzat per: Fernando Vilariño Freire, CVC
 - Curs 2018/19

1 INTRODUCTION

THIS project tackles the creation of a Computer Vision-based system that allows children with functional diversity to have autonomy of displacement within a room, through the implementation of our modular system into an existing low cost wheelchair.

The project's aims are aligned with the Goals 3 (Good Health and Well-Being), 10 (Reduce Inequalities) and 12 (Responsible Production and Consumption) of the Sustainable Development Goals [1] because of the social impact, the search of equal conditions and the philosophy of recycling and low cost. The project follows the service learning methodology [2] since the objective is to learn new knowledge in an academic way while developing a project focused on community service.

This module will allow the low cost wheelchair to autonomously move in a room, to arrive at different key points where there will be different visual stimulus. This small element of autonomy represents a huge change in the children's life since, so far, every displacement must be compulsory monitored by an adult. Especially, this will represent an enormous improvement for patients of cerebral paralysis, who may have a wide variety of unimpaired cognitive functionalities but affected motor functionalities.

In Section 2, we will enumerate the main objectives of our system. In Section 3 we will present the state-of-the-art of the project. In Section 4 we will describe the hardware system. In Section 5 we will define the proof of concept and the final system prototype. In section 6 we will describe the software system. In Section 7 we will describe our experimental set-up. In Section 8 we will show the results of the projects, in Section 9 the discussion and finally, in Section 10, the final conclusions.

2 OBJECTIVES

The main objectives of the project are:

1. Plan and create the working environment for the wheelchair and the computer vision system (scene configuration).
2. Design and implement a physical interaction device (buttons pad) for the selection of destination and its associated software module.
3. Develop a robust module for the search of the destination.
4. Develop a displacement module to reach to the selected destination.
5. Implement a small proof of concept for validation.
6. Build up a final working prototype integrated into the low cost wheelchair.

3 STATE OF THE ART

In this section we will introduce the state-of-the-art on wheelchairs for children with functional diversity, the methodology of Service Learning (particularly suited to this project), and the available Computer Vision systems.

3.1 Wheelchairs for children with functional diversity

Currently there exists some electric wheelchairs that can be controlled by the user. For example two popular models are the KuiGu Electric Wheelchair Foldable, very popular on Amazon¹, and Q700 M SEDEO PRO, which can be bought in the official page². The main problem with these kinds of wheelchairs is the price, which is between 1500 and 7000 Euros. Depending on the model, the materials and the functionalities the wheelchair will have different price. On the other hand, we cannot access to the software and change it to implement, in this case, a Computer Vision System module. This is why the low-cost wheelchair [5] that Nexe Fundació [6] offers is perfect for both reasons: as cheap as possible and access to the code.

3.2 Service Learning

Service Learning is an educational approach focused on learning during a project development which will have a social impact [3]. This project makes sense within this area for its service to the community, in this case children with functional diversity, and for its learning and research side to develop a functional system.

Moreover, the project satisfies the requirements of a Service Learning project [4].

- It contributes to the community
- It takes awareness of participating in the benefit of the community
- It obtains evidence of learning about curriculum competencies
- It shows evidence of a thoughtful process
- It participates in the adjustment of the intervention in the framework of the Service Learning.

Furthermore, as an student's project, we participated in the detection of needs, in the design of the intervention and we chose this Service Learning project having other options.

3.3 Location of the wheelchair in a room

Some authors have investigated about the use of QR codes to locate a system in a indoors space [?]. QR codes are landmarks that provide information when a system scan them. The QR code information can be filled by the user in order to give to the system useful information that will be used for instance to locate the different destination [?]. This approach is quite robust and well understood which makes it a very good candidate for our purpose.

¹[urlhttps://www.amazon.com/KuiGu-Wheelchair-Intelligent-Automatic-Lightweight/dp/B07QX7Q37R](https://www.amazon.com/KuiGu-Wheelchair-Intelligent-Automatic-Lightweight/dp/B07QX7Q37R)

²[urlhttps://www.sunrisemedical.es/sillas-de-ruedas/quickie/sillas-de-ruedas-electricas/q700-m-sedeo-pro](https://www.sunrisemedical.es/sillas-de-ruedas/quickie/sillas-de-ruedas-electricas/q700-m-sedeo-pro)

4 HARDWARE SYSTEM DESCRIPTION

The proposed system will be a module that will be added to an existing wheelchair to make it able to have autonomy. That means that the electric wheelchair without this module just can be controlled by a third person, but with this Computer Vision module, the wheelchair will be upgraded with a new functionality. The system will be implemented as a system that does not need to be connected physically to the wheelchair, the only thing needed to make it work is to plug it in the wheelchair and connect it via Bluetooth.

The system consist of the following elements (see Figure 1):

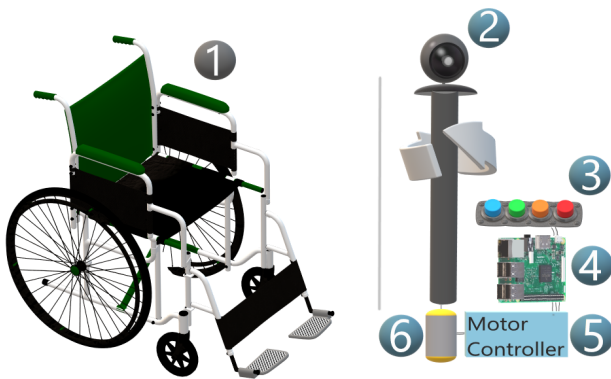


Fig. 1: Hardware elements

1. The wheelchair offered by Nexe fundació
2. A web cam
3. The buttons pad to select the destination
4. A Raspberry pi 3B+ which will run the system
5. A motor controller to control the servo that moves the wheels (only needed for the proof of concept)
6. A servo motor that will move the web cam horizontally

4.1 Hardware architecture

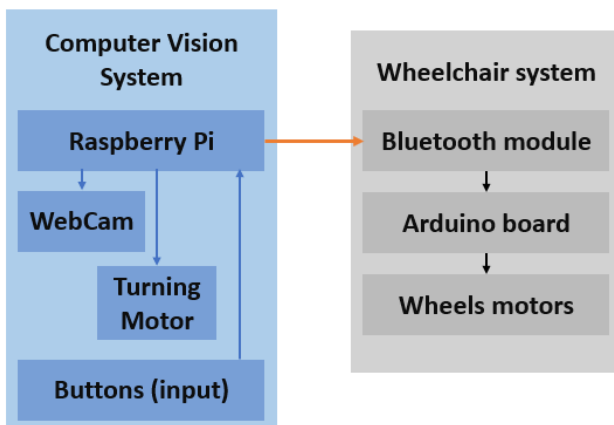


Fig. 2: Hardware diagram

The system consists of a Raspberry Pi 3B+, which will be responsible for running all Computer Vision software, which means that the system will run independently in the chair without requiring any type of connection with a server. The Raspberry Pi will have connected the buttons pad, as well as WebCam. The chosen camera is the Logitech C270 because it is very recommended for Computer Vision purposes according to some forums on the internet³. The Raspberry Pi will also be connected via Bluetooth to the arduino of the wheelchair to send the movement orders. Refer to the figures 1 and 2 to visualize the mentioned elements.

4.2 Buttons pad for destination selection

The selection of the destination will be done through four coloured buttons. The button system will be a structure created on a conventional computer keyboard. Making use of this technology gives you advantages such as the price, since you can use a keyboard that you no longer use at home or a second hand keyboard. On the other hand, the connection to the Raspberry Pi 3 is very simple, just connecting the USB cable or the wireless receptor if it is a wireless keyboard.



Fig. 3: Keyboard button structure example

As you can see in the Figure 3, when a button is pushed, the structure pushes determined keys that the system detects and knows which button is the pushed.

5 PROOF OF CONCEPT AND FINAL PROTOTYPE

5.1 Configuration of the scene

The system will work in a single room and it is designed to find four QR codes that will be located in the corners of the room.

Below is an example of placing two QR codes in a room, one placed in a corner clinging to the ends and another clinging to the wall in the plane. Once the room has all the codes properly located, the system can work

5.2 Proof of concept

In order to develop the software and all the functionalities before testing it in the wheelchair, we created a proof of concept. This proof of concept is based on a robotic car chassis with two motors for the wheels, a servo motor to move the camera horizontally, the raspberry pi, a motor controller and a powerbank. With all these components we could create a working robot to develop and test all the things that we needed. You can see a real picture of the proof of concept in Figure 5

³[urlhttps://www.chiefdelphi.com/t/best-cameras-for-computer-](https://www.chiefdelphi.com/t/best-cameras-for-computer-)



Fig. 4: Example of placement of QR codes in a room

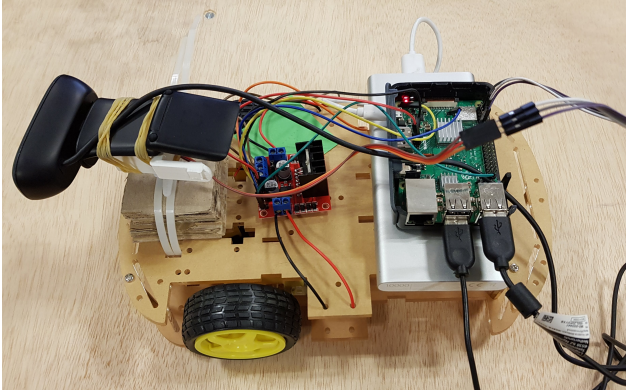


Fig. 5: Real picture of the prototype

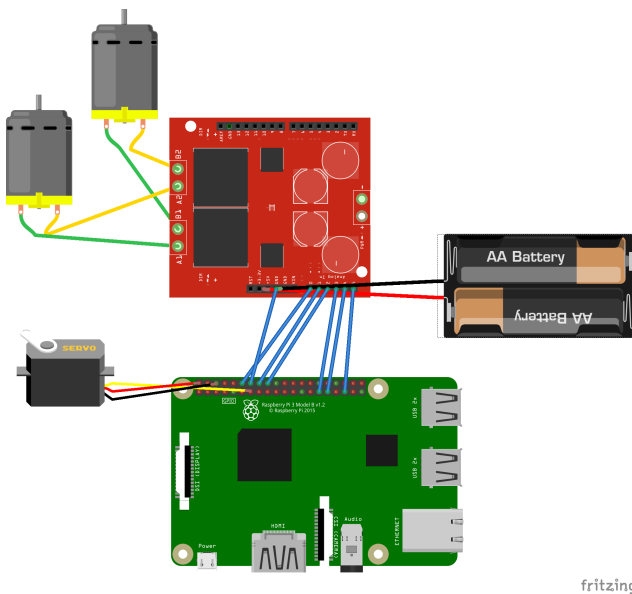


Fig. 6: Hardware connection scheme for the proof of concept model

5.3 Final prototype

The final prototype consist of the wheelchair provided by Nexe Fundació and the functional version of the Computer Vision System plugged, implemented and adapted to the wheelchair. Check Figure 7

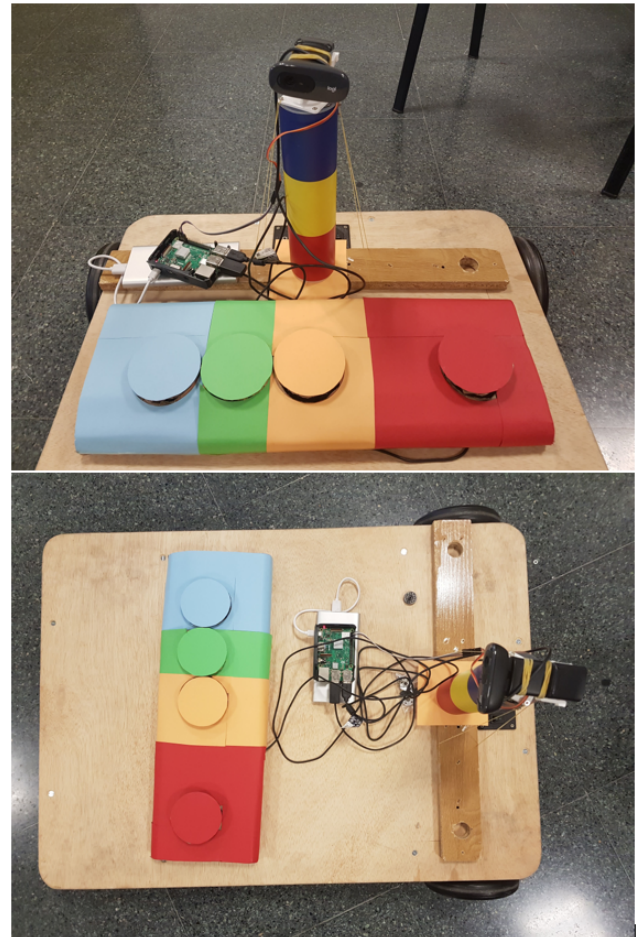


Fig. 7: Final prototype with the Computer Vision System located at the top

6 SOFTWARE SYSTEM DESCRIPTION

6.1 Software architecture

In Figure 8 you can see the software architecture of the system. As you can see, the software consist of four different modules and each module has a single finality.

- Interaction Module
- Search Module
- Memory Module
- Displacement Module

The Interaction Module waits for the user to select a destination and once it is selected, it sends the destination to the Search Module. The objective of the Search Module is to search for the destination QR code in the room. The Displacement module performs all the necessary displacements and trajectories to arrive at the desired points. Finally, the Memory Module can remember positions and information of other moments of the execution to arrive at a desired point when it is not visible. All these modules are different python libraries in order to have the functionalities divided into different files and make it easier to understand and modify. You can see the relations and behaviour between each module in Figure 8

For more information about the software system, see Annex 2: Sequence diagram of the software.

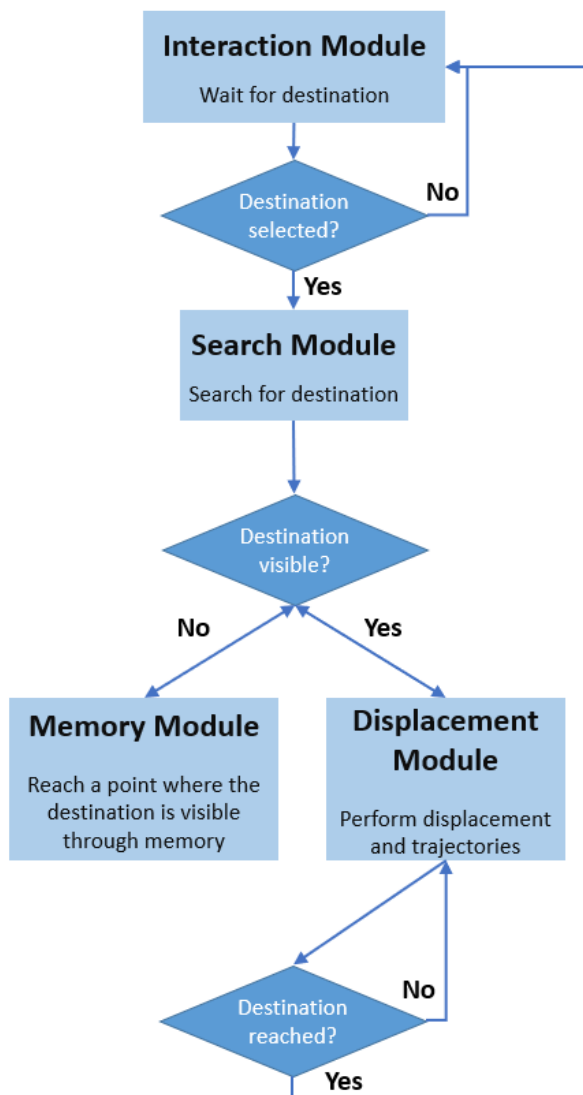


Fig. 8: Software diagram

6.2 Search module and environment information

This part explains in details how the system can know where it is through the use of QR codes.

6.2.1 QR code detection

For the detection of QR codes the pyzbar[16] library will be used. This library makes the detection of QR codes and we can obtain this information very useful to know the position of the chair. This library gets an image as an input, the image is analysed and the library give us all the QR codes detected within a variable. Of these codes we are interested in knowing the coordinates of the points that make up the four corners, the area and the total number of pixels that occupy the code within the total image.

When the library detects the points, it assigns the points depending on its position in the image, not depending on the code. This fact causes the points to change depending on the position and orientation of the code. For this reason, the code had to be modified so that the order was always the same. Therefore, the position of the points will always be:

- Up left: Point 1

- Up right: Point 2
- Down left: Point 3
- Down right: Point 4

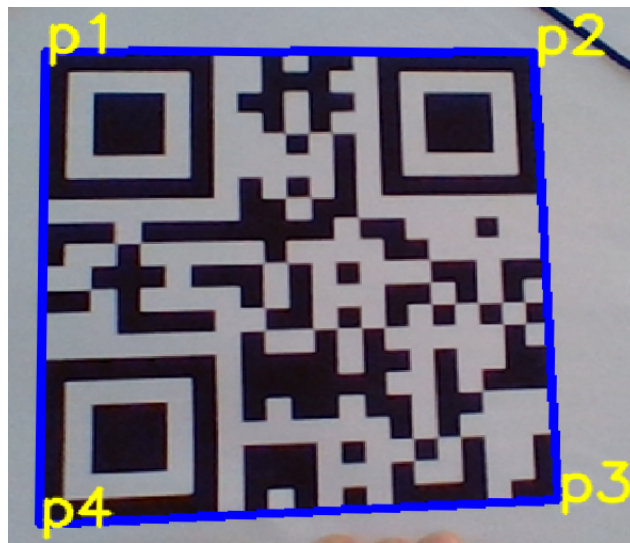


Fig. 9: Points detection

6.2.2 Camera behaviour

The camera will capture images of the room to find out what position it is. The camera will be at the top of a very stable rotating 180° axis so that the image received is as faithful as possible. The camera will turn to different position to check all the 180° and then will wait until the system turns back to check the other side of the room. This is done this way because there not exists 360 degrees servo motors.

6.2.3 QR position detection

For the detection of the position of the QR codes, the length of the lines between the points p1-p4 and p2-p3 will be taken into account. Thanks to these two lines, you can know if the camera is looking at the code from the right, the left or from the center. That is why the percentage of difference in size between the two lines will be taken into account. Thanks to using percentages, this system will always work regardless of the size of the code or its position or orientation. In total 5 positions will be detected: left 2, left 1, center, right 1, right 2 (from maximum right to maximum left in order).

In addition to this information, the distance to which the QR code is found is taken into account, since depending on how far it is, the percentages to detect the orientation will have to change. This distance will also give us more information related to the environment that will be useful to get from one QR code to another. The 'dist' variable in Figure 10 is a numerical value that varies according to the distance at which the code is found to better calibrate its orientation. The purpose of this variable is to change the difference that the two lateral lines of the codes must have since the greater distance that is found here, the smaller the length difference of the lines must be to change the orientation. This distance has been calculated doing tests manually until finding the

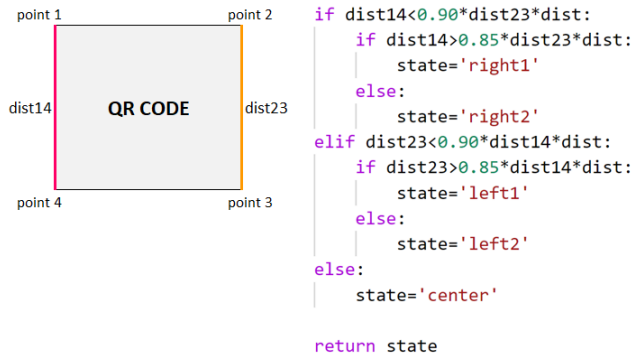


Fig. 10: Values to get the QR code orientation

best result, as well as with the position (right, center and left).

6.2.4 Detection of the destination QR

When the user has chosen the destination with the buttons, the system will check which QR codes are visible at the time of the election. If the target code matches one of the visible codes, the path to the destination will be made. Otherwise, the system's memory will be consulted to find out which direction to go in order to reach a position where the destination code was visible. The codes will have beside them a color part of one of the four available colors.

- QR-1 : Blue
- QR-2 : Green
- QR-3 : Orange
- QR-4 : Red

6.3 Displacement

This sections explains in detail how the system calculates the trajectory to arrive at the destination point after selecting it through the buttons pad.

6.3.1 Get to a visible point

When the QR code of the target site is visible by the vision system, its trajectory will be carried out following the next steps:

1. Select the destination site using the buttons. (Each code will have a color circle at its side).
2. When the target is selected, the system visualizes its surroundings and finds the code of the destination site.
3. If the chair is not looking at the code from the center and doing it from the right or left, it will be directed to the center.
4. When the system is centred in front of the code, it begins the displacement to the QR code until it is close enough. The system will do it at intervals to check that it goes in the right direction and check every time the distance to the code.

6.3.2 Reaching not visible points through the visible ones with memory

When a code is not visible, the system could reach the destination point through the memory of previous trajectories. Based on Figure 11, if the child is close to point 3 and wants to reach point 1 but this is not visible at the time of the election, the system will check what positions of the other visible points should reach to see the destination point. For example, for a previous route, point 1 was visible when the chair looked at point 2 in 'right 1' and in point 4 in 'right 1' or 'center'. Thanks to this information, the system can already know to which direction the wheelchair should go. In addition, the distance to the codes will be taken into account so that the route is as safe as possible. At the moment the chair has reached a point where the destination code is already visible, it will perform the normal path to this point.

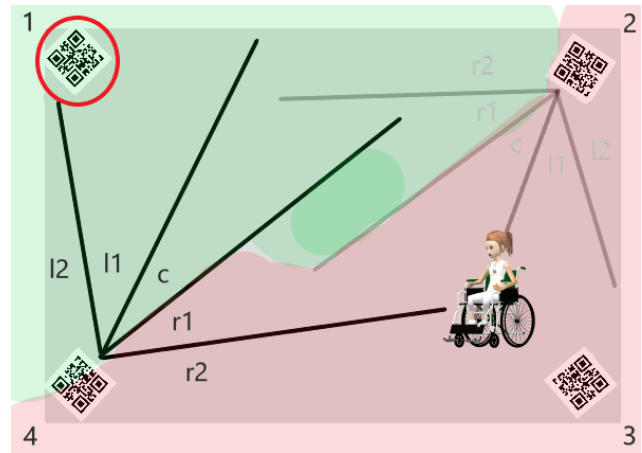


Fig. 11: Visibility map to arrive to QR code 1

When the system starts, after selecting the desired destination, it checks its surroundings to get all the necessary information for the memory module. So, if the system starts in the center of the room, where all the QR codes are visible, it will know how to arrive always at this point if sometimes does not find the destination.

The memory is an array than contains all the QR codes and some information that the system sees when checks its surrounding at the beginning. Then, if the systems can see all the four QR codes, the memory will be:

```
[[1,distance,orientation], [2,distance,orientation], [3,distance,orientation],[4,distance,orientation]]
```

Otherwise, if it only sees the QR codes 2 and 3, the memory will be:

```
[[2,distance,orientation],[3,distance,orientation]]
```

where distance is the distance between de QR code and the camera (based on the size of the QR code) and the orientation is from where the system is seeing the QR code (right2, right1, center, left1, left2).

When the user selects a destination and it is not visible, the system performs the following steps:

- When searching for the destination, check which other

QR codes are visible.

- Once the system knows that the destination is not visible, check which of the visible codes was closer to the system when the memory was filled.
- Go to the destination based on the selected point to be able to see the final destination point
- Once the desired point is reached, check again for the final destination point.
- Once the final QR code is detected, move to the point as usual.

7 EXPERIMENTAL SET-UP

7.1 QR specifications

For a good detection of the codes, these must be placed on a flat surface, such as a wall. It is very important that the code is flat, since any alteration or curve in the code can make its detection more inaccurate. The code is recommended to be attached to a harder surface, such as a cardboard, a wood sheet, polyester or similar. The use of this material is mandatory if you want to place the codes in the corners of the room, since with this mechanism the four corners of the code can be clamped better in the wall and it will continue to show itself flat. It is advisable not to plasticize the codes since the plastic can create reflections and may make it difficult to detect the codes. It is also mandatory to put the codes at the same height as the center of the camera.

The QR code must be created using on-line code creation tools[13]. In these tools you can choose the message that will be displayed when the code is detected. This message must be a number from 1 to 4. The size of the codes must be 600x600 to 1000x1000 pixels printed on a DIN-A3 sheet with margins, so that its detection is the best possible. Therefore, once the code is printed, the size of its sides must be 26 to 28 centimetres, without including the margins.



Fig. 12: QR codes used for the experimental set-up

7.2 Variables and measures used in the proof of concept

As you can see in Figure 13, the position of the servo depends on the pulse value that is receiving. For the proof of concept the used values are the ones that are in Figure 13, where 500 is maximum left and 2500 is maximum right.



Fig. 13: Pulses values for the servo motor on the prototype

On the other hand, in Figure 14 there are the values that the motor controller is receiving depending on the situation of the system. Notice the speed of the motors depending on the movement, which motors are working in every case and the action that will be done after performing the movement.

Motors	Move forward	Stop motors	Move Left	Move Right
Speed	80%	0%	65%	65%
Motors A	forward	-	forward	backward
Motors B	forward	-	backward	forward
Finish			Move forward and turn Right	Move forward and turn Left

Fig. 14: Motors behaviour for the different movements

7.3 The button pad implementation

We have created the structure using cardboard and very cheap materials to make it as cheap as possible. Check the image of the final result of the first version in Figure 15

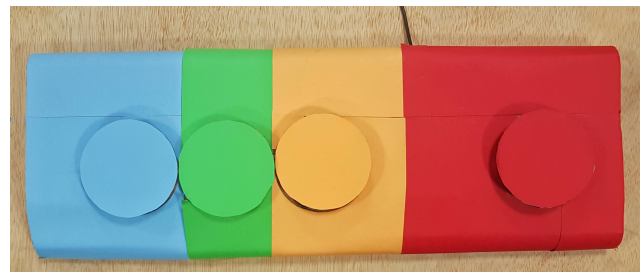


Fig. 15: Buttons pad first final version

See Annex 1 for more pictures about the creation process of the buttons pad.

8 RESULTS

Furthermore, in Figure 17 you can check graphically that when the distance in meters gets lower, the size of the qr code in the image from the camera is bigger.

In terms of results, the computer vision part is finished and fully working. The system actually detects all the QR codes that is seeing and it knows where to go according to the destination selected by the user. The software has been implemented, changed and tested in the prototype to check in real life how does the system works. Thanks to that, both the Computer Vision System and the prototype are able to perform the required actions. The user can select a destination through the button pad very simply thank to the colours of the buttons and the colour part in the QR code, as you can

• Key	Button 1 a,w,s,x	Button 2 t,h,g	Button 3 p,o,l	Button 4 1,5,4,8
----------	---------------------	-------------------	-------------------	---------------------

TABLE 1: KEYS AND BUTTONS RELATIONS.

	Meters	QR size (px)	QR size (%)	
Low distance	1,18	113220	9,24	<ul style="list-style-type: none"> Low distance Ideal distance High distance
	1,35	80316	6,53	
	1,47	65750	5,33	
Ideal distance	1,65	53550	4,35	
	1,85	43442	3,53	
	2	37627	3,06	
	2,2	32578	2,65	
	2,35	29240	2,38	
	2,6	23254	1,91	
High distance	2,9	18632	1,51	
	3,5	12768	1,05	
	3,8	10920	0,88	

Fig. 16: Distance values and the relation with the QR codes information

see in Figures 12 and 15. The system is able to scan its surroundings in order to fill the memory data to us if it does not find the destination. When the destination is selected, the system will search for the destination and arrive at the desired point in the approximate times that can be seen in the Figure 18. The system check first on the front and if it does not find the destination, it turns back and checks again. To scan, the servo motor moves the camera to different positions according to the pulses shown in Figure 13. When the destination is detected, the robot turns to watch the destination directly and starts moving to the point. If the system is detecting the QR code from the extremes (for example, it is viewing the code from very far to the right), it will go to the front of the code to start getting closer to the destination viewing the code better and in a safer way. If the system loses sight of the destination QR code, it will scan again to find it and continue the route. After scanning the second time for a QR code, if it is not visible, the system will use the memory module to arrive at a point where the code should be visible. The system is checking all the time if it is going straight to the QR code to focus to the code if it is necessary. Finally, when the system reaches the destination, the system waits for another destination through the buttons.

So, at this point of the project, these are the elements that are already developed and working.

- A Computer Vision System that has all the functionalities to give autonomy to an existing electric wheelchair.
- A functional proof of concept that works and it's good to test and improve the Computer Visions System.
- A destination input system through a reused keyboard that works thanks to a buttons structure. This part is completely finished and works perfectly.
- A memory module that can help the system to arrive to destinations that are not visible in specific situations.

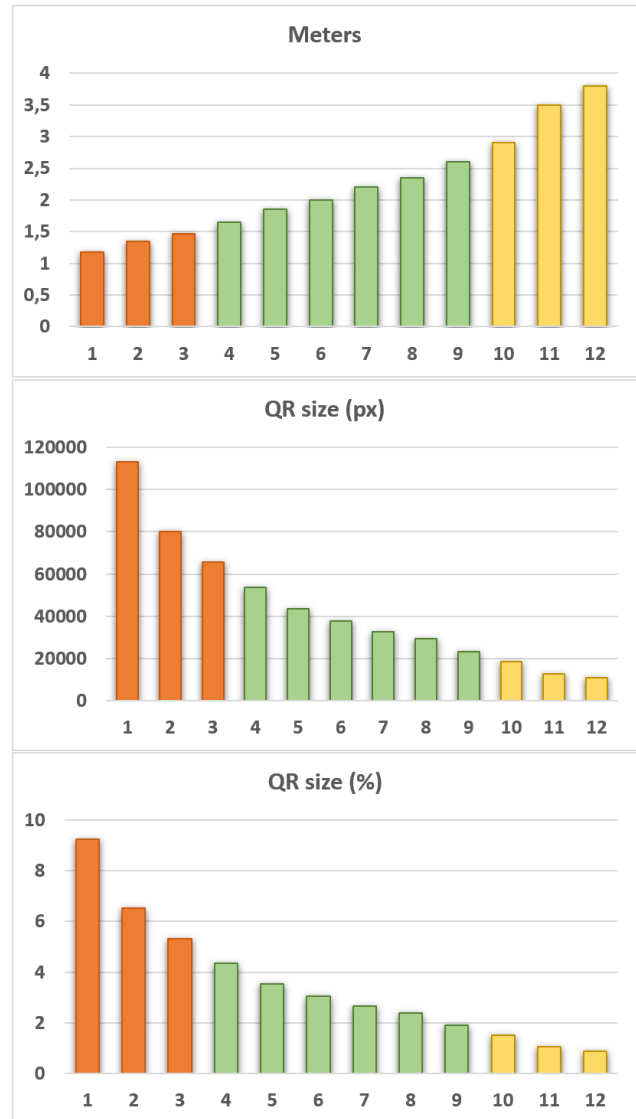


Fig. 17: Graphic of the distance values and the QR codes information

- A system prototype with the real wheelchair and the project implemented on it that works and have autonomy, but that can be improved by analysing and adapting both the wheelchair code and the Computer Vision System code.

There are differences between the proof of concept and the prototype. In the proof of concept everything is working, all modules. Despite the cheap materials and the instability of the system in some movements, the proof of concept works as expected. On the other hand, in the prototype, the interaction module is exactly the same, so it works perfectly. The search module works but there are differences in terms of time spent in the rotations to adapt it to the wheelchair. The same happens with the displacement module, it works but some parameters have been changed to adapt them to the chair. The memory module depends on parameters from the Search and Displacement modules, so the behaviour is not as good as in the proof of concept. As you can see in Figure 18, the proof of concept is faster in the scanning because it is a smaller system and more stable for the web cam, this is why it can take pictures more quickly. Otherwise, the displacement is faster in the system

prototype because it has more powerful motors and it can move faster than the proof of concept.

In Figure 16 you can notice the relation between the distance between the camera and the QR codes in meters and the information that the system is getting from the QR codes to know how close it is, the size of the QR code in pixels in the image and the percentage that is occupying of the total of the image. You can also compare the values with which the system works correctly. When the distance is within the red range, the system has reached the destination. So the system can work without trouble in big rooms because it detects codes that are almost 4 meters away.

Time (in seconds)	Proof of concept		Prototype	
	Minimum time	Maximum time	Minimum time	Maximum time
Scan (180°)	10	14	25	30
Scan (360°)	24	32	57	67
Reach destination	30	90	15	40

Fig. 18: Time spent by each system in performing the search and displacement actions

We have uploaded a video of the wheelchair performing the search of the destination and the displacement⁴

9 DISCUSSION

Based on the purpose and the main objectives of the project and the final results, there is enough evidence and good results to conclude that the result of the project has been successful. All the main objectives and some improvements of the original concept have been fulfilled, such as:

- Implement the system to work as a module than can be plugged very simply in the wheelchair
- Design and implement a low cost button pad that follows the philosophy of low price and social purpose of both projects, the wheelchair and the computer vision system.
- Implement a proof of concept to test and develop the system rather than asking for a real wheelchair.
- Implement an optional memory module to make the system more intelligent and efficient.

According to the results mentioned before, we come to the conclusion that the system can work without trouble in big rooms because it can detect QR codes that are almost 4 meters away. This distance is good enough to accomplish the purpose of the project. On the other hand, the time that the wheelchair spend in searching and reaching the destinations is quite good, but it can be improved by checking and modifying the wheelchair's code to make it more adapted to our system and creating a more stable base for the web cam. With all these points fulfilled, it is concluded that the project has been successful and that it opens up to be improved to become a successful system that helps children with special needs giving them more autonomy and security in order to make them happier.

10 CONCLUSIONS

Looking back to the main objectives of the project and comparing with the results, we can say that we have a system that satisfactorily satisfies the children's need to move autonomously through a room thanks to a very robust search module that cant locate the wheelchair among all the QR codes to finally arrive at desired point in a short period of time. On the other hand, the buttons pad is very useful and fulfils its objective the whole time, in addition to being a very economical and solvent solution. One of the strongest points of the project is that these functional system works as a unique module that can be plugged in the wheelchair very simply without having to modify any intern thing, just place it and connect it through bluetooth. Moreover, the user will feel very comfortable in the wheelchair using this system because the movements are smooth and the execution times are very reasonable.

11 FUTURE WORK

The next steps are continue working on the integration of the system in the wheelchair in order to make it as comfortable as possible for the children. The memory module can be improved as well to save more information of different moments to use it when necessary. Also we will work on the implementation of a system that work with images instead of QR codes in order to make the system more user friendly. To finish, we will try to get a 360 degree servo motor to make the system more comfortable and we will try to adjust the parameters more for a perfect operation.

12 ACKNOWLEDGEMENTS

I like acknowledge to Nexe Fundació for making us get the idea of the need to make this system and for trusting us to develop it and make it possible. I would also like to thank you for having left us the original wheelchair to be able to implement the system, as well as offering us all the information we have needed throughout the development.

I would like to thank David Garcia for giving us all the necessary information about service learning and introduce us into this way of learning while carrying out projects with social impact. Thank also my tutor, Fernando Vilariño for helping me during all the project and for his enthusiasm and dedication. Seeing your enthusiasm about the project has made me more eager to do my best and your advice has made me take the project to another level.

Para finalizar me gustaría agradecer a mi familia por su apoyo constante durante el desarrollo del proyecto y durante toda la carrera. Me ayudais a plantar las semillas de todas mis ideas y me acompañais mientras los frutos crecen. Esto es gracias a mi hermana Cristina, mi madre Cleopatra y mi padre Joaquín.

⁴<https://youtu.be/VqULQrx76ow>

REFERENCES

- [1] Sustainable Development Goals
<https://www.un.org/sustainabledevelopment/sustainable-development-goals/>
- [2] UAB - Aprentage Servei (Service Learning)
<http://pagines.uab.cat/aps/ca>
- [3] McMillan, Janice and Goodman, Suki and Schmid, Barbara; Illuminating “Transaction Spaces” in Higher Education: University–Community Partnerships and Brokering as “Boundary Work”
Journal of Higher Education Outreach and Engagement
<https://www.rs-online.com/designspark/lidar-radar-digital-cameras-the-eyes-of-autonomous-vehicles>
- [4] Romero, David Garcia and Lalueza, Jose Luis; Procesos de Aprendizaje en Aprendizaje-Servicio Universitario: Una Revisión Teórica
Journal of Higher Education Outreach and Engagement <https://www.rs-online.com/designspark/lidar-radar-digital-cameras-the-eyes-of-autonomous-vehicles> 2019
- [5] Silla exploradora
<http://www.sillaexploradora.es/>
- [6] Nexe Fundació
<http://www.nexefundacio.org>
- [7] Youtube - How Google’s Self-Driving Car Works
<https://www.youtube.com/watch?v=YXylqtEQ0tk>
- [8] Waymo; Youtube - Waymo 360° Experience: A Fully Self-Driving Journey
https://www.google.es/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=2ahUKEwjHoNeowdvgAhXFxoUKHfRoDDoQwqsBMAB6BAgFEAQ&url=https%3A%2F%2Fwww.youtube.com%2Fwatch%3Fv%3DB8R148hFxPw&usg=AOvVaw0rR_mTMhPAiv7Nl5kLiKmo
- [9] Bill Marshall; Lidar, Radar and Digital Cameras: the Eyes of Autonomous Vehicles
<https://www.rs-online.com/designspark/lidar-radar-digital-cameras-the-eyes-of-autonomous-vehicles>
 February 21, 2018.
- [10] Daniel Lynn Lerner, Jared Stephen Russell ; Method and system for determining and dynamically updating a route and driving style for passenger comfort (Google Patents)
<https://patents.google.com/patent/US10107635B2/en>
 August 19, 2016
- [11] Adrian Kaehler, Gary Bradski. *Learning OpenCV: Computer Vision with the OpenCV Library. (English)* O’Reilly Media, September 24, 2008.
- [12] Jan Erik Solem. *Programming Computer Vision with Python: Tools and algorithms for analyzing images. (English)* O’Reilly Media, June 29, 2012.
- [13] QR code generator online tool
qr-code-generator.com
- [14] Mattheieu Napoli; Looking for ways for a robot to locate itself in the house
<https://stackoverflow.com/questions/6520570/looking-for-ways-for-a-robot-to-locate-itself>
- [15] Wikipedia - Field of View / Camp de visió
https://ca.wikipedia.org/wiki/Camp_de_visio
- [16] PYZBAR, Python library
<https://www.pydoc.io/pypi/pyzbar-0.1.6/>
- [17] Best cameras for Computer Vision
<https://www.chiefdelphi.com/>
- [18] Car chasis, motor controller and cables links - Amazon
amazon.es/car/chasis
amazon.es/motor/controller
amazon.es/cables

Annex 1: Images of the creation process of the button pad

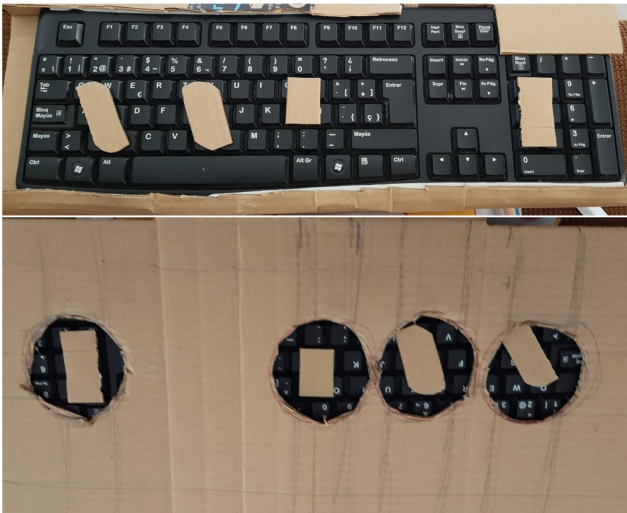


Fig. 19: Keyboard button structure process example

Annex 2: Sequence diagram of the software

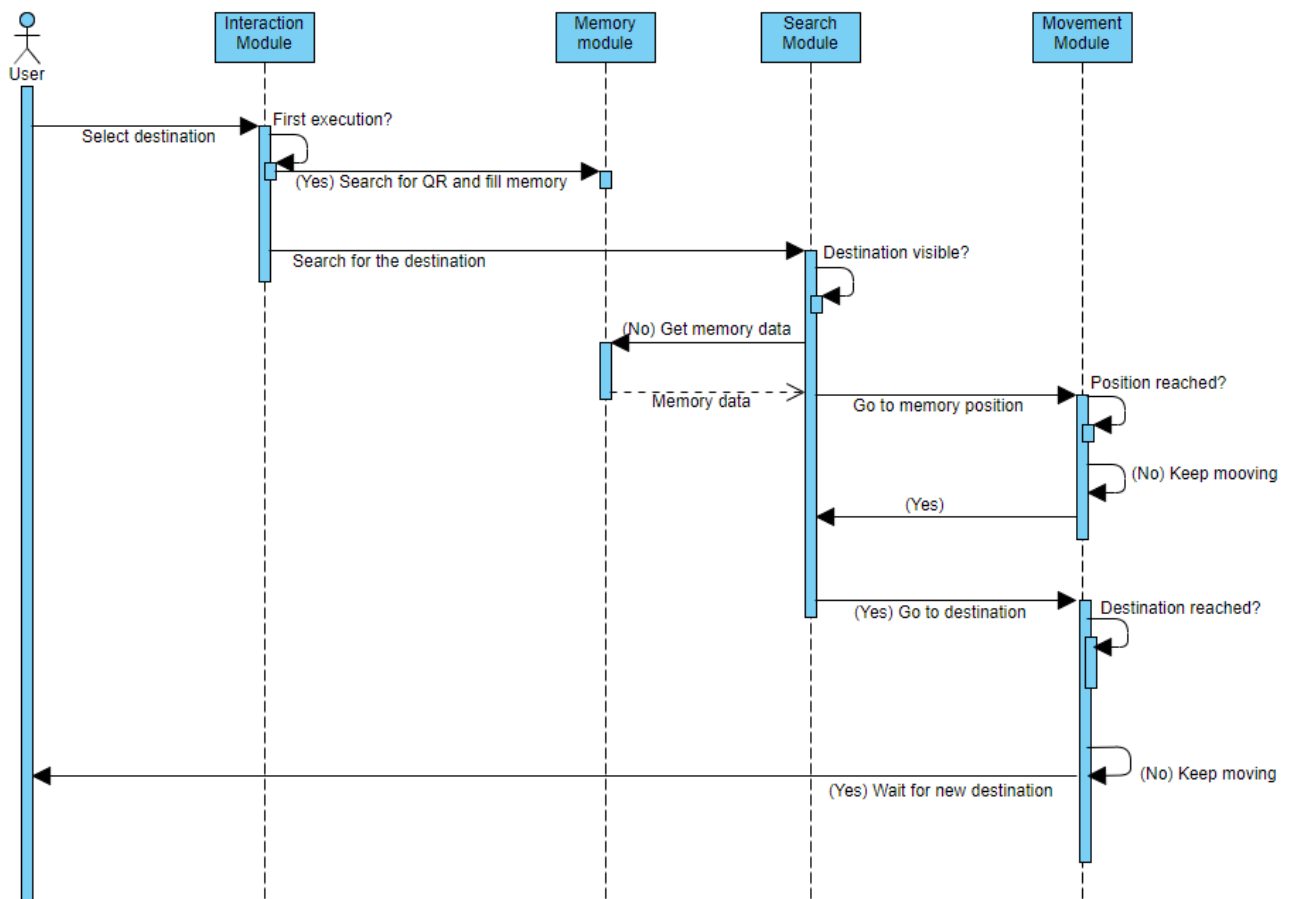


Fig. 20: Sequence Diagram of the software