Università Politecnica delle Marche

Master's Degree in Biomedical Engineering

Analysis and development of robot on board sensors network to localize people in indoor living environments

Supervisor:

Prof. Ing. Gian Marco Revel

Co-Supervisor:

Dott. Ing. Sara Casaccia

Dott. Ing. Nicole Morresi

Candidate:

Ilaria Ciuffreda

Academic year 2019/2020

To my Family. Thanks for being
an inexhaustible source of love,
support and joy.

Contents

List of Figures

List of Tables

Acronyms

AI = Artificial Intelligence
AAL = Active and Assistive Living
GPS = Global Positioning System
RFID = Radio-Frequency Identification
LIDAR = Light Detection and Ranging
RGB-D = RedGreenBlue- Depth image
KCF = Kernelized Correlation Filter
RiSH = Robot integrator Smart Home
PIR = Passive Infrared
DBN = Dynamic Bayesian Network
LRF = Laser Range Finder
VGG-N = Visual Geometry Group Network
PF = Particle Filter
VLP-16 = Velodyne-16
IR = Infrared
SLAM =Simultaneous Localization And Mapping
CV = Computer Vision

Chapter 1

Introduction

1.1 Motivation

In recent years, technology has become so important in our lives that today living without it seems unthinkable. We are surrounding ourselves with increasingly intelligent furniture that are able to interact with us. Equipped with artificial intelligence (AI), these devices can improve the quality of life of the people who inhabit that environment. In this regard, the trend that is increasingly being observed is that of inserting social robots that integrate sensors and AI into the living environment. Social robots are increasingly used especially in the AAL (Active and Assistive Living) environment for the care of the elderly. It is precisely this aspect that has increased the interest of researchers due to the pandemic period of COVID-19. In fact, the pandemic has increased the need to assist the elderly at home and create innovative services through technology that can be installed in the home. A robot within a home environment can be programmed to assist seniors in their daily activities and to monitor people in need of care. For this reason, the possibility of having a fool proof system for human detection in an indoor environment could lead to numerous improvements in everyday life. Indoor human monitoring can be used in smart homes to study the needs of inhabitants based on their location and movement. For example, there are intelligent devices on the market that can automatically adjust the lights or the heating system near the person; by combining this technology with an audio system it is possible to follow the user as he moves inside the building. All this can lead to a decrease in energy loss by turning off some electronic devices when the user is not close to them. Indoor location system can be used to activate an alarm system when the presence of intruders is detected in the house. This system could also be used inside museums where the user's position could be used to provide contextualized content on the artwork he is looking at through the audio guides [1]. A better use of this technology can be the indoor human localization at home to improve the life quality of people with health problems giving them the possibility to live a more dignified life. In addition, this

technology can improve the quality of life of caregivers, since this system could be the "eyes, ears and communication channel" for homecare nurses.

These aspects are turning into reality thanks to the GUARDIAN project, of which this thesis is an integral part. GUARDIAN project is an EU-founded project that proposes the deployment of a senior friendly social robot that can be accessed remotely by the nurse and the informal careers.

Given this background, it is understandable how the localization of the person in the indoor environment can represent a practical help in making life easier, more comfortable, smarter and even safer. Considering all these advantages, the growing interest in this field is understandable.

1.2 Objectives

The overall objectives of this thesis are to identify on a robot, available on the market, the sensors able to track and localize a human in an indoor environment and to build an algorithm capable to perform this task. Furthermore, the measurement accuracy of distances returned by sensors chosen in localizing and tracking the user, is analysed.

To achieve this goal, the on-board robot sensors have been used, without including additional sensors at home sensors network. To this aim, this work is organized as follows:

- Understanding the potential of the robot chosen for this aim
- Exploring all the sensors on the robot and decide which one is useful for our purpose
- Implementing an algorithm for person localization and tracking in an indoor environment using the chosen sensors
- Performing an experimental validation of used sensors.

1.3 State of Art

1.3.1 Misty Robotics

Robot story begins roughly 60 years ago [1] at a time when hardly anyone knew what a computer was, let alone what to do with one. The creation of the first stable hardware platforms to evolve the robotic industry, offered to developers a place to build the applications that would ultimately make computers essential to everyone. The robotics

industry story is very similar to the ones of other technologies, e.g. smartphones. In order to reach its potential, robotics developers needed a stable and open platform to make robots useful for everyone, but this is what was missing. That stable platform that did not exist until a couple of years ago is now available thanks to Misty Robotics. The goal at Misty is to solve the hardest problem facing robot industry building a baseline robotics platform that makes programming robots easier even if one never worked with robots before. The problem with robotics is that there is no existing home robot that anybody can modify. Misty is addressing that issue by building essentially the canonical robot platform. Misty skills can be anything: they can be created in software; they can be hardware-based or they can be a combination of the two. Misty Robotics goal is to provide a stable platform for developers equipped with the latest and greatest features like mapping, object recognition, face detection, passage of touch voice, recognition path, planning and automatic charging. When all these things come together with AI increasingly cheaper microcontrollers can be built, creating in this way the perfect storm for the physical creation of extraordinary thing like companion robots. The first version released by Misty Robotics was commercialized in 2018 with the name of "Misty I". Misty I robot was a very basic version, and it presented a lot of limitations. In 2019 the company released an enhanced version of Misty I, "Misty II", with a completely new design and some new amazing features giving rise to an open tool robot that can be used for servicing the ageing, disabled people and education for children.

Misty II has been chosen to develop the work of this thesis and in the next chapters it will be explained how is possible to exploit its functionalities to perform a human localization and tracking in indoor environments.

1.3.2. Sensors for human indoor localization

Lots are the solutions proposed by researchers on sensors that could be used to accomplish the task of localization and tracking of a human person in indoor environment. A very common technology that is used in external environment but that cannot be applied in closed environments is Global Navigation Satellite Systems (GPS OR GNSS). GPS relies on radio transmissions in the spectrum of microwaves (on frequencies close to 1500 MHz) but suffers a heavy absorption phenomenon when they have to pass through roofs, walls and other conductive objects, features that make this technology useless inside buildings [2].

To go beyond those limits and allow the usability in a domestic environment new technique have therefore been studied by the scientific community.

In [3], Zhihua W. et al. suggest the classification of sensors according to the type of technology deployed to determine the position:

1) **Inertial sensors** widely used for their highly accessible equipment but with the drawback of having a low accuracy and for this reason used in many cases as auxiliary of the acoustic localization sensors.

2) **Radio frequency (RF) sensors** characterized by a sub millimetric accuracy but subjected to interferences with electronic devices and they include solutions based on laser beam or on radars.

3) **Vision sensors** which, differently from RF, makes no interferences to ubiquitous RF based device because they use an optical information acquired by an omnidirectional camera or a three-dimensional camera. The drawback of these sensors is linked to privacy problems.

4) **Acoustic sensors** which can achieve a relatively high accuracy and low time latency, but they suffer from sound reflections, which limits their absolutely accuracy.

5) **Infrared sensors** which can use both natural and artificial light whose spectrum differs from visible light. This technology is unobtrusive for humans compared with indoor positioning technologies based on visible light.

Some researchers proposed robot on-board sensors, others proposed sensors installed in the home environment and wearable technologies. In this work a description of above-mentioned sensors is performed to give to the reader an overview on these systems and to provide a critic analysis to understand which sensor is convenient to use to accomplish the final aim. It is essential to select a suitable sensor for elderly care among the existing ones, to classify the performance metrics to evaluate the available system. In this thesis are analysed:

1) **Reliability, Precision and Accuracy**: considering that the system should be used in a home environment the accuracy of these sensors must be much less than the size of the room of a house. Additionally, the system must be good in quality and in performances, giving as precise information as possible on the human position in the room.

2) **Safety and Security**: the system must not use devices that over time may compromise the health of localized persons (safe). Moreover, the information on the position of the

person or the presence or absence of people inside the room must be protected against external attacks by malicious people (secure).

3) **Easiness , Intrusiveness**: considering as target user an elderly person not perfectly capable to use complex electronic devices, the system should be as simple and intuitive as possible so that, after the installation it is able to run in background without any specific user activities. Most indoor localization systems are based on tags use that the user must carry around for be located. In general these tags are based on very common and easy to use electronic devises like smartphones and smartwatches but that they turned out to be uncomfortable and reluctant for user who must wear a device in every moment of the day, especially in the relaxing moments. For this reason, the passive and tag-less localization systems are considered the most suitable for the scopes of this work because these systems should not interfere with the user daily activities and movements.

4) **Privacy aware**: the system must be able to protect the user's privacy even in environments where the user does not want to be filmed. For this reason, if an image acquisition system for human localization is chosen, the user must be informed in advance. On the other hand, whoever builds the system must ensure that the images acquired are correctly encrypted and obscured.

5) **Cheap and easy to install**: to be available to the higher number of users, the system must be as cheap as possible, and it should not require high installation costs.

6) **Low maintenance and exploitation cost**: a localization system must not have a high maintenance cost and the user must not spent too much to keep it on.

In real life, it is difficult to find a system that is a good compromise among all these specifications. A brief description of the more significant and recent (from January 2016 to May 2020) papers present in literature is proposed. Firstly, the methods that involves the use of a single sensor are studied and then the techniques which require a home sensor network installation.

1.3.3 Single-sensors based method

A. Radio-frequency sensors

Several localization methods use radio frequency to localize people in indoor environments. Some authors proposed solutions based on radar while other proposed solution based on laser. In [4], Palopoli at al. proposed a human indoor localization

method based on Radio Frequency Identification (RFID). This technology is less intrusive and can be used for device-free identification. The main drawback of this type of system is that it requires the use of a separated transmitter, installed in a home environment, and receiver which typically is the robot.

Another solution for tracking humans is proposed by Alvarez A. et al [5]. Their methodology uses a laser beam built using a Laser Imaging Detection and Ranging (LIDAR) to keep a continuous identification of human's legs in time and space to detect individuals. Even if this is a very efficient method, Lidar is too expensive for home use. A solution to this issue is proposed by Zhao et al [6] with the introduction of a millimeter-wave radar. This innovation tends to overcome the use of Lidar sensor and of RGB-D camera, given the elevated cost of the first one and the limited tracking range and accuracy of the second one. Their study was the first one in which the point cloud generated by a millimeter wave radar are used to track and identify people while they are walking. Even if this method achieved great success reaching high performances, a big drawback is represented by the inability of this sensor to track simultaneously more than two people.

B. Vision-based sensors

The use of video cameras for monitoring people inside a room is one of the most traditional and popular methods. The work presented in [7-8] based the human detection and tracking on an RGB-D camera. More in details, the authors in [7] present an efficient and accurate detecting and tracking algorithm for a moving target for a mobile robot equipped with an RGB-D Camera and a laptop. An efficient modified Kernelized Correlation Filter (KCF) is applied as visual tracker. The KCF is based on the idea of traditional correlation filter and it uses special metrices to significantly improve the computation speed. Information obtained from depth camera is used to calculate 3D position of the target and provides environmental information for the robot to decide its path. The system has great performance on the detection, tracking and following, and the system is able to move with avoiding obstacles if there are some in the scene. The authors in [8] instead proposed a method like the one proposed by [7] but in this case the RGB-D camera not only tracks a human but also follows the human after that its localization an environment, thanks to the use of a pre-installed environmental map. However, when the target is moving too fast with changing speed the system is not perfectly worked for the mobile robot following part.

Authors in [9] used two digital cameras, as vision-based technology, with a fixed distance between them and an efficient working space of about one meter. This approach results to be effective and feasible to apply in various industrial approach, but the drawback is the use of two cameras increase the cost of the entire system. Similarly, authors in [10] presented a stereo camera, i.e. a camera able to capture a 3D image, as solution to perform people tracking. The results show how this sensor is suitable and effective for the problem at hand even if it is not tested on a robot and with multiple targets.

In summary, camera-based monitoring systems are very versatile and can be used for several purposes. This system can distinguish the different users by carrying out specific actions according to the case. Drawbacks of this system regard the energy consumption, cost-inefficiency, high computational costs, and lack of privacy. As mentioned above in case you decide to use this system it is of fundamental importance that the user is aware that his privacy may not be completely protected even if the manufacturer has installed encryption protocols.

C. Acoustic-based sensors

The human localization using audio sensors is performed considering the user as a sound source. Using more than one microphone is possible to detect the exact position of the user in a room through the triangulation method [11]. Accuracy and reliability of the system can be improved by increasing the number of microphones.

The localization can be performed with a centimetre-scale accuracy in a quite inexpensive way and without interfering with user's daily routine. The main drawback of this system is that the audio signal can interfere with other audio signals causing a false user localization.

D. Infrared-based sensors

Dongning et al. [12], in their paper proposed an indoor multiple human targets localization and tracking using thermopile sensor. The sensor, named GridEye, is not installed on a robot, but it is located on the ceiling. The thermopile is used to localize and track multiple human targets and to collect data of human motion. An adaptive threshold method is used to remove non-human thermal targets and background image. Considering GridEye's technical specificationsand the viewing angle effect, the human targets can be localized. and can be associated with trajectories by using target positions and their heat

signatures. This sensor has a good accuracy while the resolution of the GridEye sensor is 8 by 8. This latter can be increased to 71 by 71 using an interpolation algorithm and a Gaussian Filter to smooth the expanded array. Drawbacks of this system are due by its inability to detect a specific user because not capable to define the profile of a person and by its limited sensor's field of view.

1.3.4 Multiple sensors based methods

Ha Manh Do et al. [13] proposed a robot-integrated smart home (RiSH) which integrates a home service robot, a home sensor network, a body sensor network, a mobile device, cloud servers, and remote caregivers. The RiSH is able to perform a human position tracking, fusing the information that comes from a PIR (Passive InfraRed) network with information from IMU sensor through the Sequential Monte Carlo algorithm. Moreover, using a DBN (Dynamic Bayesian Network) model, the robot is able to recognize the human fall sound and consequently to activate itself. The DBN is a Bayesian Network that include a time constrain among the all the considered variables. Hardware design on the robot include the installation of a Laser Rangefield (LRF) vision system, an auditory system, an Intel NUC minicomputer and batteries. The human must be equipped with a physiological sensor (Pulse Oximeter, Textile Electrodes, Respiration Belt) motion sensor installed on the right thigh (Smart Garment) and a smartwatch. Home sensors are the PIR and the GridEye sensors. The accuracy of all these sensors is extremely high and the coverage area is from 18 to 433 inches. Drawbacks of this system are that the map of the room must be known, and the complexity of the entire sensors network. RiSH system provides human body activity recognition with an accuracy > 86%, while the human location tracking is performed with a mean square error lower that 0.2 m.

Chao J. instead proposed a cooperative human localization system that uses a mobile robot and smartphones to localize moving person. Human localization is performed using a robot on-board Kinect sensor combined with smartphone based acoustic ranging subsystem. An extended Kalman Filter-based dynamic positioning algorithm is developed and integrated with the acoustic relative subsystem to provide real time localization of the moving human target [14]. The median accuracy ranges from 0.43 m to 1.12 m under different environmental noises and different human walking speeds. The drawback of this localization system is that human has the necessity to have a smartphone on the hand, condition that cannot be satisfied by some categories of elderly.

A more attractive approach that exclude the use of a smartphone for the localization is proposed by S. Yang et al. [15]. They proposed a method based on the use of multiple Kinect cameras to localize the human in an indoor environment. The choice to use multiple Kinect cameras is justified by the high accuracy, the low interference between multiple cameras, and the better occlusion handling that this combination of sensor presents in comparison with the use of a single Kinect camera. Thus, the use of a calibrated multiple Kinect sensors permits to scan a large area and to track and record the skeleton data of a human body. Then, a classification model is able to recognize the daily activities and determine the amount of time spent in an area of the house. A drawback of the presented method is the ability to track only one person per time in a room.

In [16] Wang M. et al, proposed a new method for tracking the 3-D positions of a person by monocular vision and ultrasonic sensor. This system is tested sequentially and independently in indoor and outdoor environments with a robot platform. Visual tracking processes videos captured from the camera sensor to estimate the target's locations in the image coordinate. Ultrasonic array sensors offer the range information of the target in the robot coordinate; the actual 3-D positions are estimated by merging these two heterogeneous information sources. The accuracy of this system is relatively high thanks to the use of a visual tracking algorithm for indoor and outdoor environments. Another advantage of this system is that it overcomes problems of occlusion, scale variation, missing targets and re-detection. It represents an alternative to more complex and costly 3-D human tracking systems for mobile robots. Drawbacks are associated to the use of a camera RGB because, as mentioned before, it has some privacy problems and, additionally it is impossible to detect a human if he is partially occluded. In the study proposed by Halimaa [17] the RGB-D camera is fused with the thermal sensor. This idea was yet proposed by [18] but in [17] is included the velocity of the head in the state vector to improve fast motion reducing the number of false alarms. The fast motion tracking also in an occluded condition is obtained using particle filter (PF) algorithm based on head position. For each depth-thermal image pair, the head position is first segmented in the depth image, and then matched with the thermal image using calibration information to predict the actual position according to the previous state. The fusion of thermal and depth information is used to update this predicted state. The combination of this sensors revealed to be efficient permitting to reach higher accuracy respect to the previous method [18]. Authors like [19] proposed instead a method for human activity recognition by

combining different modalities from RGB-D sensor. In their work, dynamic images trained on pre-trained VGG-F network and depth images captured by different angles. each of these depth images are trained separately on pre-trained VGG-F network. at the end, a combination of them is performed. The results were obtained by testing the method on different datasets and achieved great performances. In [20] instead is presented a human-robot interaction procedure, designed to collect motion trajectories of people in a generic indoor social setting with extensive interaction between groups of people and a robot in a spacious environment with several obstacles. The locations of the obstacles and goal positions are set up to make navigation non-trivial and produce a rich variety of behaviours. The participants are tracked with a motion capture system; furthermore, several participants are wearing eye-tracking glasses. In addition to the video stream from one of the eye tracking headsets, the data includes 3D Lidar scans and a video recording from stationary sensors. Abhijeet Shenoi et al. in their work [21] try to overcome the problem of the RGB camera associating it to a LiDAR sensor. On one hand, 2D RGB images allows us to discern appearances of objects to effectively identify and classify them even at large distances. On the other hand, 3D point cloud data is sparse but due to the depth information it allows to discern objects which might overlap in a 2D RGB image but are well separated in 3D space. To obtain real-time 3D tracks, integrates recent state-of-the-art solutions for 2D detection in RGB images and 3D detection in point clouds, uses novel multi-modal descriptors, and improves upon well-established data-association and filtering techniques. The accuracy is increased thanks to the use of the RGB camera, but the privacy problems still remain. Sidagam Sankar et al. [22] implemented a real-time human detection and tracking on a wireless controlled mobile platform fusing the information coming from a RGB-D camera with the information coming from a point cloud. The results showed that the user can easily use a cell phone to remote control the mobile platform for the detection and tracking of multiple humans in a real-word environment. In [23] instead the Microsoft Kinect sensor is used with a Hokuyo laser sensor to perform a human following with a mobile robot. The tracking methods implemented are face detection, leg detection and person blob detection. The results showed a higher efficiency for human following (74.29%) on leg detection method. Similarly, Redhwan A. et al. [24] proposed a human following of a mobile robot using colour features extracted using the hue-saturation value histogram. In [25] Yan et al., instead proposed the human tracking and localization based only on a 3D LiDAR. Their

system is composed of four main components: a cluster detector for 3D LiDAR point clouds, a multi-target tracker, a human classifier, and a training sample generator. The cluster detection is able to detect clusters of point clouds, which are passed into a multi target tracker and a human classifier. The first one permits to tack a human cluster in 2D plane estimating the horizontal coordinates and velocity, while the second one is able to classify human and not-human basing on 10 parameters. The last component, the sample generator is able to increase the accuracy in the detection through the use of two experts: P-expert converts false positive into true negative while the N-expert works in opposite way, converting the false positive into true negative. Respect to the previous studies their solution is able to track human in large indoor environments and the training of the neural network is based on an online learning which is able to increase the adaptability of the robot to different environments. The accuracy of the system is increased respect to the state of art as the cover range. In addition, the cost of this device is low. The sensor used in [26] is the Velodyne LiDAR's Puck sensor. Velodyne's new VLP-16 sensor is the smallest, newest, and most advanced product in Velodyne's 3D LiDAR product range. Vastly more cost-effective than similarly priced sensors, and developed with mass production in mind, it retains the key features of Velodyne's breakthroughs in LiDAR: Real-time, 360°, 3D distance and calibrated reflectivity measurements

1.3.5. Summary of the State of Art

A summary of the methods described in the previous sections and their main characteristics is reported in Table 1.

Table 1. Comparison between the most used Indoor positioning systems based on single sensor. Data inside the table have been taken by different articles cited in this chapter [3-12].

SINGLE	SENSOR	BASED	METHOD				
Methods	Accuracy	Price	Safety	Tag	Privacy	Power	Notes
Vision-based	high	high	yes	no	no	high	High resolution but lack of privacy and high cost
Infrared sensors	20 cm	low	Yes	no	yes	low	Privacy aware, cheap but hot sources can give faults
Sound	33 cm	low	yes	no	yes	low	Cheap but other sources of sound different from the person can interfere
Radio-frequency	high	high	yes	yes	yes	medium	Accurate but not able to distinguish people

Table 2:Comparison between the most used Indoor positioning systems based on multiple sensors. Data inside the table have been taken by different articles cited in this chapter [13-26].

MULTI SENSOR BASED METHOD							
Methods	Accuracy	Price	Safety	Tag	Privacy	Power	Notes
Vision, Sound, Infrared	High	high	yes	no	No/yes	high	Extremely complex system and a map of the environment is necessary
Sound, Vision	mediu m	high	Yes	no	No	low	Require the constant use of a smartphone by the user
Multiple vision	high	high	yes	yes	no	high	It is necessary to provide a calibration of the cameras
Ultrasound Vision	high	high	yes	no	no	low	Privacy problem and the user cannot be detected when he is partially occluded
Vision, Infrared	high	high	yes	yes	No	medium	High computation cost and time consuming
Vision, Radio-Frequency	high	medium	yes	no	no	medium	High accuracy gives by the use of Lidar but this is an expensive system

Among the systems analyzed, the vision-based system is the one capable of achieving very high accuracy and distinguishing users in an indoor environment. The biggest drawback could be the lack of privacy. A system that solves the privacy problem is the ultrasound sensor system but requires a tag on the user making this system intrusive.

By analyzing these systems in terms of costs, the cheapest ones are the one based on a technology that the user adopts in its daily life, such as the smartphone or the RSSI which uses a Wi-Fi connection system. The first method is inopportune as it requires that user always carries the smartphone with him, while the second method could cause a false location as the device could interfere with other devices that transmit on the same radio frequency. The method based on human sounds are useless because full of errors caused by the interference with other sound sources. The other systems analyzed require the construction of a very complex and very expensive network of sensors. Even the maintenance of these systems could be very complicated, as a malfunction of a single sensor could cause damage to the entire network. The infrared sensor instead needs an installation phase, but no large upkeeps are necessary. A major drawback associated with this type of sensors is that over time they can negatively impact on user's health since it is constantly subjected to infrared emissions.

1.4 Related work

GUARDIAN project belongs to the AAL (Active and Assisted Living) programme started on 1st of January 2020 which seen ten participants organization came from Italy, Netherland and Switzerland work together to build a social robot companion to support homecare nurses.

This project introduces a social companion, which aims to be of direct benefit for three groups of end-users:

(1) The GUARDIAN social robot will be the “eyes, ears and communication channel” for homecare nurses.

(2) Informal caregivers need their work and want to support their loved ones. They experience high levels of stress and mental and physical fatigue as they worry about them and have difficulties finding a work-life balance due to the additional care tasks. GUARDIAN provides a helping hand.

(3) Frail seniors prefer to live in their own house for as long as possible. Informal caregivers and homecare nurses cannot be present 24/7. With GUARDIAN, the seniors have a companion that can take over tasks from caregivers and supports them in prolonged independent living.

The communication between the robot and the frail person should be performed through dialogues and messages as naturally as possible. The robot can be used to keep track of daily activities of the person and can be used to remind to the frail person to perform same daily task such as eating or taking medicine. In addition, the robot should be able to understand if someone is nearby in order to start a conversation or to remember a past one.

GUARDIAN will be developed according to an iterative methodology with three streams which are co-creation, ethics and the business modelling. The final GUARDIAN eco-system is mainly described in Figure 1.

In this thesis an algorithm for the track of the person during daily activities is implemented.

To accomplish this task a vision-based sensor is chosen. The privacy problem of this sensor has long been analyzed until a compromise has been reached. The method proposed in this thesis acquires a photo only when the localization process is initialized. For tracing, the user's face markers saved during the image acquisition phase are used, minimizing the number of images acquired and limiting the privacy problem.

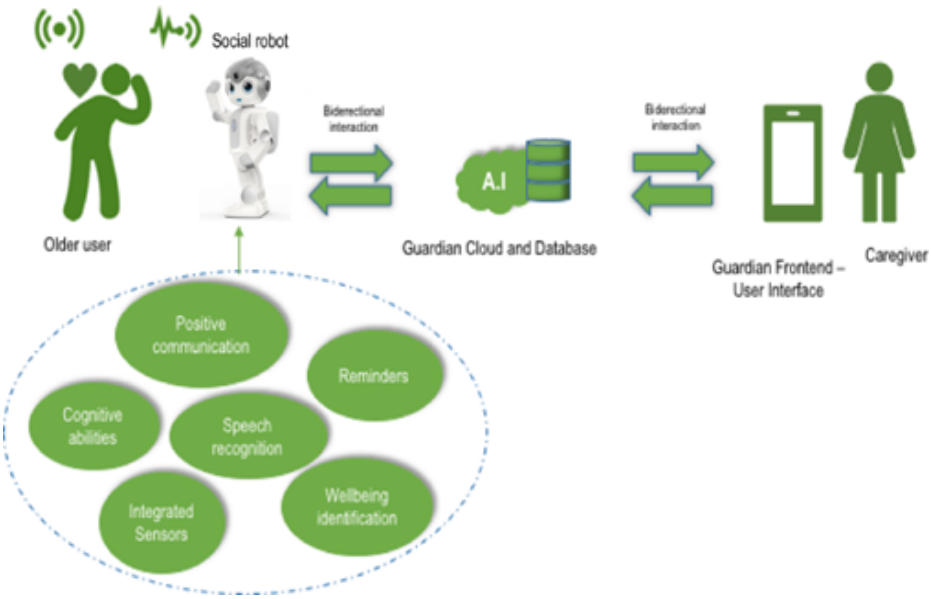


Figure 1: graphic representation of GUARDIAN ecosystem.

In the following chapters every aspect of the implementation of this algorithm will be explained more in detail. In particular in chapter two the materials and method for the implementation of the algorithm are described. In chapter three the results of the chosen method are presented and in the chapter four the results are discussed. In chapter five the conclusion on the chosen method are reported and an overview to the future technologies that we are studying to improve the localization and tracking of a person is presented.

Chapter 2

Misty II onboard sensors

2.1 Introduction

Misty II has more than 25 sensors which include 6 capacitive sensors, 8 time of flight sensors, 10 bump sensors and one occipital structure core depth sensor.

The capacitive sensors are considered not specific intended use sensors but that does not mean they are not important. There are four sensors on Misty head, one sensor under her chin and one sensor in the carrying handle in the back of her head. These sensors are proximity sensor that detects nearby objects by their interaction on the electrical field created by the sensor. Misty's capacitive touch sensors use 6 pieces of thin foil under the plastic shell connected to a controller. When a conductive object (e.g. a finger) comes close to the foil, there is a capacitive coupling between the foil and the object which is detected by the controller. Misty's capacitive touch sensors are a simple way to let people do cute, clever, and more intimate things. They are low power yet provide a useful mechanism for her to react to human contact in funny and human or animal-like ways [27].

The time of flight sensors tell Misty that she is about to fall when she comes to close to an edge or they tell Misty that there is an object around her. When one of these two conditions occurs, time of flight sensors act stopping Misty motion at 0.15 meters from the dangerous situation. There are three forward time of flight sensor, one rear and four on the edge/downward. Time of flight sensors permit Misty to move confidently in the environment. To the time of flight sensors are linked the bump sensors that instead reveal a collision between Misty and objects. There are 10 bump sensors. Three of them tied in parallel on each front corner and two tied in parallel on each rear corner.

The occipital structure core sensor is the most expensive sensor and it is located on Misty's head. This sensor is used for 3-D map creation and for the facial recognition skill. It includes a 166° diagonal field of view wide angle, dual IR cameras and an IMU. This

structure also houses a 4K RGB camera (Figure 1). In the following subchapters the sensors installed on the occipital structure core sensor will be described.

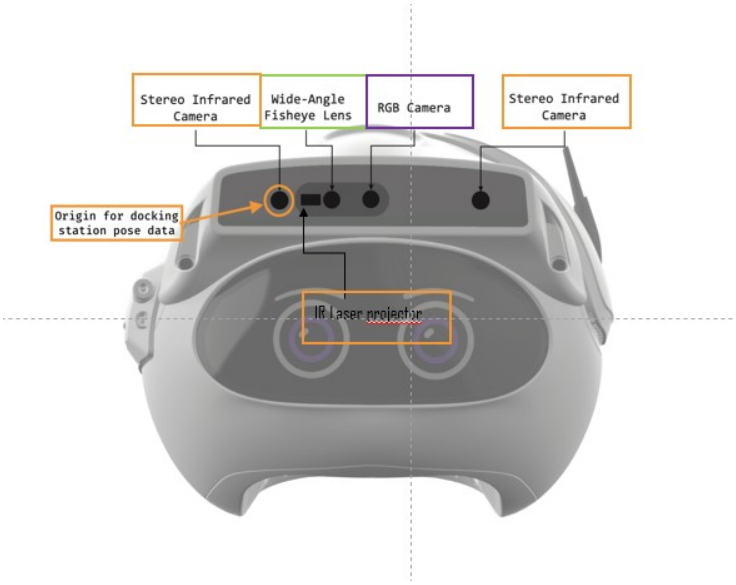


Figure 2: Occipital structure core sensors. In orange are indicated the sensors used for the 3D depth image reconstruction; in purple are indicated the sensor used for the face detection, face recognition and face training; in green is indicated the sensor used to augment the tracking of where Misty moves. The combination of this last one with the 3D depth image sensor is used to create a map and know where Misty is within the map at all the times.

2.2 RGB camera

One of the aims of this work, aimed at exploring all Misty's sensors to provide an evaluation on which is the most appropriate to accomplish to the final aim.

In this section the 4K-RGB camera is studied. In particular, the optimal distance at which a person's face must be, to be captured by Misty, is evaluated. This analysis is done by taking a photo with the 4K sensor and processing it on a modified algorithm for face detection.

2.2.1 Experimental Protocol for human tracking

To understand the optimal distance at which the person can be detected using an image collected from the 4K camera, it was necessary to set-up an experimental protocol.

There are 4 parameters evaluated on 25 pictures captured by the camera. These parameters are the distance human-robot, the pitch angle of robot head, the human size that should be detected, the number of people present in the environment and the environmental condition, i.e. the presence of object in front of the human that could make inefficient the human detection.

The human detection on the captured pictures was performed using a modified YOLO algorithm. YOLO (You Only Look Once) algorithm is a smart convolutional neural network (CNN) for doing object detection in real-time. The algorithm uses a single neural network for the input image, then it splits the image into areas predicting bounding box and probabilities for each one. The modified YOLO algorithm was devised to recognize only people and not on object. To understand the best condition at which is possible to detect a person using the 4K camera, in the experimental protocol a classification of distance human-robot in four measurement ranges is performed. The ranges have been selected as follows:

- 0m-1m
- 1m-2m
- 2m-3m
- >3m

Pitch angle of robot head is divided in three sub-angles:

- small angle: 0°-5°
- medium angle:6°-20°
- high angle:21°-29°

A prototype of the table used in the experimental protocol is showed below (Table 3).

Table 3: Prototype of the experimental protocol used for evaluation of optimal human-robot distances

DISTANCE HUMAN-ROBOT			PITCH ANGLE OF ROBOT HEAD			ENVIRONMENTAL CONDITION			HUMAN SIZE		ACCURACY	HUMAN DETECTION	
0 m-1m	2m-3m	>3 m	Small	Medium	High	object	room size	n. of peaple				YES	NO

2.3 Depth Image

Depth image is acquired using two stereo infrared camera and an infrared (IR) laser projector. Misty II does not directly provide the raw image, but a reconstruction of the raw data is necessary. As described in Algorithm 1, the reconstruction of the depth image was performed using Python and the script built to do this is divided in two parts. In the first part a vector reshaping is performed to obtain a 240x320 size image. The second part is used to plot the reshaped vector. In figure 2 is showed a depth image extracted from

the sensors, while in Figure 3 is showed one of the best extracted images among all those processed.

Algorithm 1 Depth image reconstruction

1. // depth image acquisition
2. **input:** http request to robot for depth acquisition
3. **output:** vector with 74800 values
4. // reshaping code
5. **input:** vector with 74800 values
6. **if**
7. **output:** reshaped vector with a size of 240x320

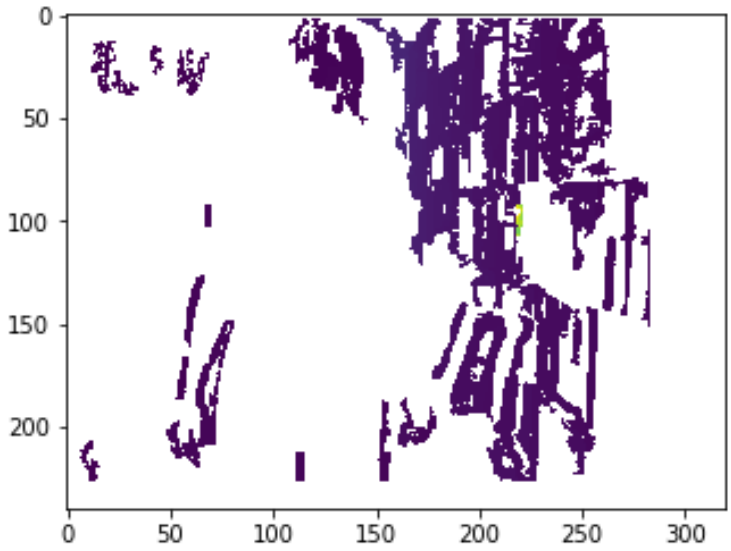


Figure 3: example of depth image reconstructed using Python

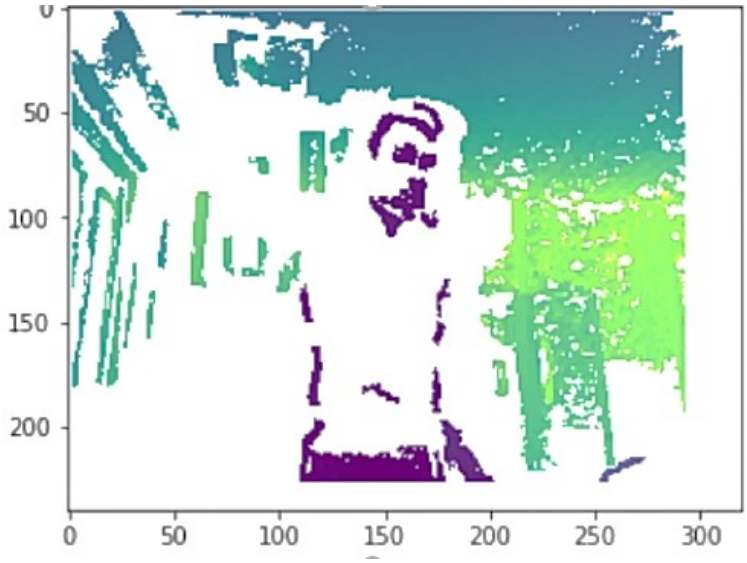


Figure 4: image acquired using depth sensor. This is one of the best reconstructed images

2.4 SLAM algorithm

The SLAM (Simultaneous Localization and Mapping) algorithm is activated when an environment reconstruction is performed or when a location of robot inside the map is required. A robot must be able to navigate in its environment in a secure and unfailing way. That means a robot needs to reach its goal avoiding static and dynamic obstacles. Depending on the type of map representation different approaches to SLAM problem can be used. The most common approaches are based on occupancy grid and landmark-based map. The latter perceives the environment as geometrical features giving rise to a very compact map. A drawback is that this approach is based on predefined features extraction making necessary to know in advance some environmental structures. Instead, occupancy grids can represent high-handed structures, in which the environment is discretized into cell that can be free, occupied, or unknown. On one hand this type of SLAM approach is advantageous for its high accuracy level if an appropriate resolution of the grid is used, but on the other hand this approach requires a huge amount of memory to store the data. Occupancy grid can be two-dimensional (2D) or three-dimensional (3D) [28]. A 3D map reconstruction compared to a 2D map reconstruction requires more specific sensors (vision sensor can be used for example) and takes even more memory.

In Misty II, the SLAM sensor uses an occupancy grid approach. The algorithm receives data from two IR cameras, the IR laser and the wide-angle camera used to increase tracking of where Misty moves.

These sensors emit laser rays in some predefined directions and receives their reflections to give us the distance. Rays travel longer distances if the objects are far away in their directions; otherwise rays travel short distances when reflected from objects nearby. The robot collects this information over time while moving around. Anything hit by the laser's rays appears bright. In contrast, places where the rays hit unobstructed area appear dark in the figure. To understand how the occupancy grid map from laser reading is built some terms must be defined. The term occupancy is defined as a random variable. Random variable is a function from a sample space to the real. In this case occupancy is defined the probability that space has two possible states: free and occupied. The occupancy random variable then has two values: 0 and 1. An occupancy grid map is just an array of occupancy variables. Each element of the grid can be corresponding with a corresponding occupancy variable. Occupancy grid mapping requires a Bayesian filtering algorithm to maintain an occupancy grid map. This algorithm requires a recursive update to the map. Because a robot can never be certain about the world, the probabilistic notion of occupancy is used.

In a map point of view there are two possible measurements:

1) a cell could be passed through by the ray which means it is free empty spaces

2) a cell could be hit by the ray which means the cell is occupied by something.

The zero is used for the free measurement while the 1 value is used for the occupied measurement.

If we think to a probabilistic model of the measurement giving the occupancy state of each cell, there are only four possible condition probabilities of measurements that we can enumerates, because the variable z and m are all binary. The probability that z is one, given m is 1 is the probability that we have occupied cells. Probability that z is zero and m is one is the probability that we have free measurement for an occupied cell. We can give the probabilities on observation giving m is 0 in the same way. These are the measurement parameters we need to send. False measurements stand from sensor noise, moving objects and uncertain knowledge of the robot motion.

The occupancy variable that represents the state of grid cells and the measurement model parameters that will be used to update the map. We want to update the probability of each cell from our measurements in a Bayesian framework. However, keeping track of probabilities directly can be hard. Instead of using occupancy probability cell a new concept that will make the computation simple can be introduced. If there is the probability that something happens it can be written as p ox x then the odds can be considered as the ratio. This ratio is the probability of the thing happening over the probability of the thing not happening.

Applying the Bayesian Rule, we can rewrite the odds to include the sensor model term, the prior term for both, the numerator, and the denominator. If we take the logarithm of both sides of the equation, we have that the left-hand side includes the posterior odds and the right-hand side includes the sensor model and the prior. The resulting formula represents the log odds update of the occupancy grid mapping. The map stores the log odds values of each cell and the measurement model is represented as the log odds as well. A computation for mapped updates becomes editions of those log-odds. There are two things you need to remember when this updates rule is applied

1) the updates are done only for observed cells.

2) the updated values become priors ‘values when a new measurement in future time steps is received.

The update rule becomes recursive. Figure 3 and Figure 4 represent how discussed until now.

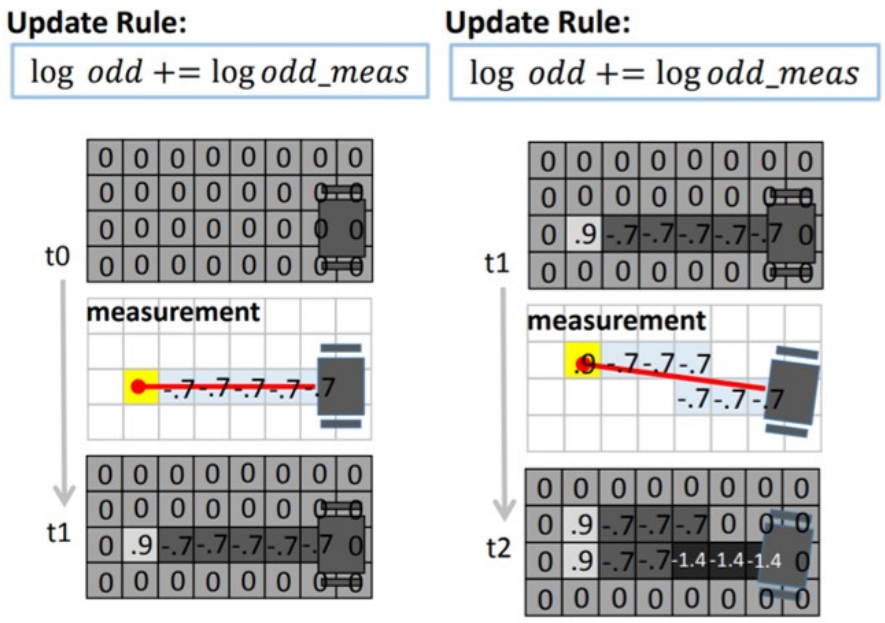


Figure 5:example of how the discretised map cells are update using the update rule

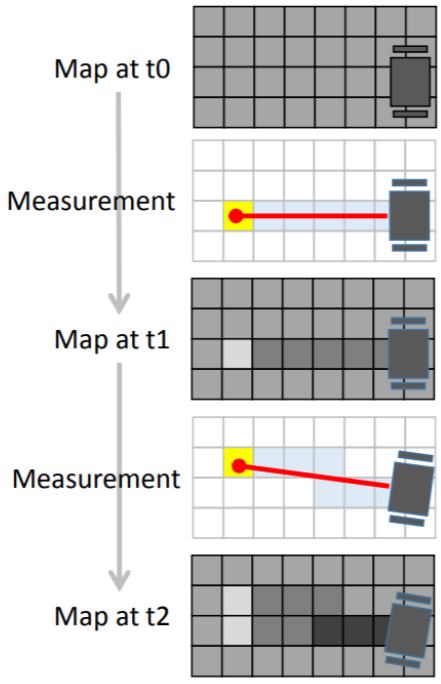


Figure 6: example of how the occupancy grid map is built over time

Considering known the robot pose on the map, another problem regards the transformation of the sensor output data from the robot reference system to the reference system of the global map.

The origin of the coordinate frame in the map is expressed in Figure 5 and it is on the down-left corner. The moving robot has a coordinate frame described in figure 5. Considering that the laser sensors emits a ray along the x coordinates and that the sensor receives a distance measurement D, it is important to define which is the cell hit by the ray. To explain how this occurs, a continuous version of the map is first analysed and then this is converted into discretized map. Assuming that the location of the robot and its direction are known, we can place the robot into the map like shown in figure 5. To find the coordinates of the obstructed point, we use the distance measurement and the known pose of the robot. A continuous representation for the position can be used. However, the map we are going to build should be represented as a discretized cells over certain resolution. The location on the grid is expressed by the formula:

$$i = \text{ceil}\left(\frac{x}{r}\right) \qquad \text{Equation 1}$$

Then, it is necessary to compute the indices of free empty cells for which a specific algorithm (Bradenham's line algorithm) is used. Basically, the algorithm takes two points as input arguments and returns a list of cells that forms an approximation to a line segment between the two input points.

In a more realistic setting, the sensor has more than one ray emitted in different direction. Let's say that the sensor emitted five rays each one returning a distance respect to a body frame location. If the pose of the robot is known the position of the cells hit by the rays can be computed in the same way but in this case the direction in its ray is taken into the computation (Figure 6) . The algorithm of Bradenham can be used for induvial ray to find three empty cells and then take the union of them [29].

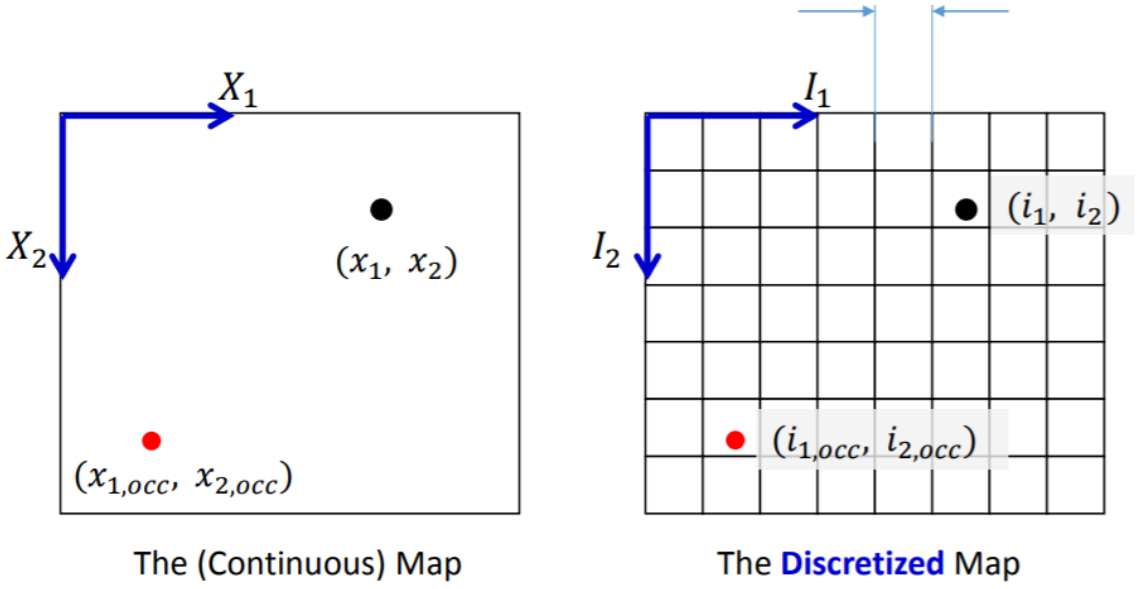


Figure 7: example of how the position of cells hit by rays is reconstructed from the continuous map representation to the discretized map

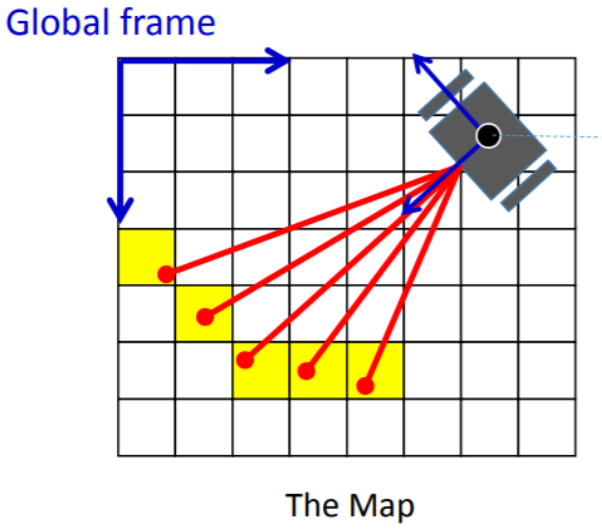


Figure 8: example of how multiple rays emitted by the robot

To perform an environmental reconstruction, it is necessary to move Misty inside the room by sending a command via the Command Center [1] available on the Misty robotics website. Misty saves all the reconstructed maps locally and you can switch between maps by sending an appropriate request to Misty's REST API endpoints. Once the preferred map is selected Misty can get her pose (position and orientation) within the map, mapping herself every time her pose changes. To deduce the Misty's pose, both the odometry and the observations of the environment are used. The odometry defines the relative robot

pose between two successive points of time. And the information about the environment is given by sensors like a laser range-finder. In Misty, the activation of a laser range-finder is done activating the tracking command. In a known environment, the uncertainty of the robot's pose is low because each observation of the environment can be compared to the known map. But in an unknown environment, the uncertainty is larger. The odometry is not perfect and the error increases over time. To reduce this error, the information of the sensors needs to be compared to the map, but the map also contains uncertainty because it is not a-priori known.

After the robot obtains the pose, using the SLAM in the navigation module it is possible to move Misty autonomously from one point to another, creating a point-to-point path. To view the Misty map using Python you need a reconstruction algorithm (Algorithm 2) capable of converting the raw data into an image. The rebuild algorithm consists of exporting the raw map data from the robot by sending to Misty an HTTP request and reshaping the raw data. The raw data graph shows an image with four colours representing the open area, occupied area, covered area, and unknown area. The map scale is expressed by an output value, the "meters per cell" value, obtained after map creation.

The following image shows an example of a Misty II map obtained from the command center (right) and a reconstructed one (left).

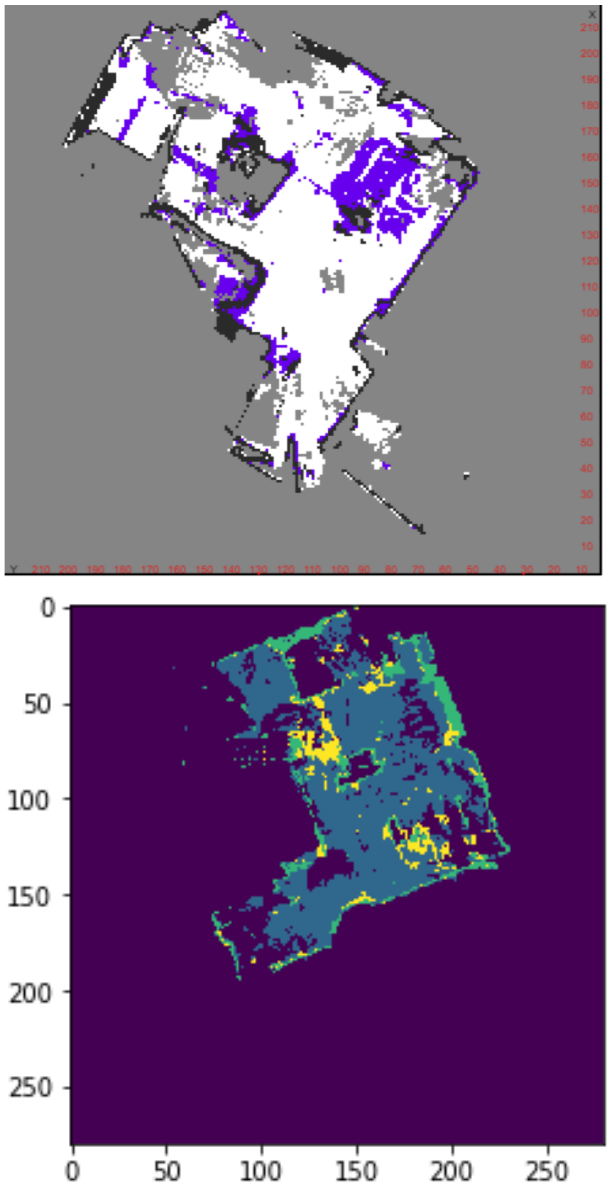


Figure 9 : On top: the map obtained from Misty command center. On bottom: the map obtained from a row data Python reconstruction

[1.] The Command Center works by sending requests to Misty's REST API endpoints.

As visible in Figure 3A the map refence system is on the right bottom corner while in Figure 3B the map reference system is on the left top corner, meaning that every time you want to pass from a map representation to another one a coordinate change is necessary.

Algorithm 2 Map reconstruction

1. // map information
2. **input:** http request for last map acquired

```
3. output: name of last map acquired
4. //check if the name of the last map is saved on laptop
5. input: last map name
6. if
7. output: true
8. extraction of grid raw data, height and width values of map and meters per cell value for map reconstruction. A reshape of grid value is done according to the height and width values and map plot.
9. else
10. save on laptop grid raw data, height and width values of map, meters per cell value. Map plot.
```

2.5 Face Recognition and Face Detection

Face recognition and face detection features are part of the computer vision (CV) block that runs on Misty onboard processors. Face recognition algorithm uses the 4K camera to detect and recognize faces using Misty's visor. Misty uses a dedicated CV module that runs locally on the Snapdragon Neural Processing Engine on each Misty II robot, so that it can be invoked even without internet connection. Face recognition implies a preliminary step: in fact, before Misty robot can recognize people, it must be trained on their faces. To do this, you can use either her onboard API or Misty's API Explorer. The latter allows you to get started quickly if you do not want to handle this by building your own application. Once Misty learns your face, Face Recognition can start.

The CV module also offers another feature which is the Face Detection. In this one the operating principle is the same as the facial recognition skill, but the difference is that Face Detection does not require a prior training process. From one point of view, this can be considered an advantage because it allows you to save time in identifying a person, but on the other hand for the purposes of this thesis it represents a disadvantage since in the tracking not only the person on whom you want to activate the tracking is identified but other people present in the environment are considered.

According to the previous consideration in this thesis a human tracking using face recognition is proposed. To build a script for the Face Recognition able to give us information on the distance at which a person is detected, a WebSocket connection from Misty robot and Python is established.

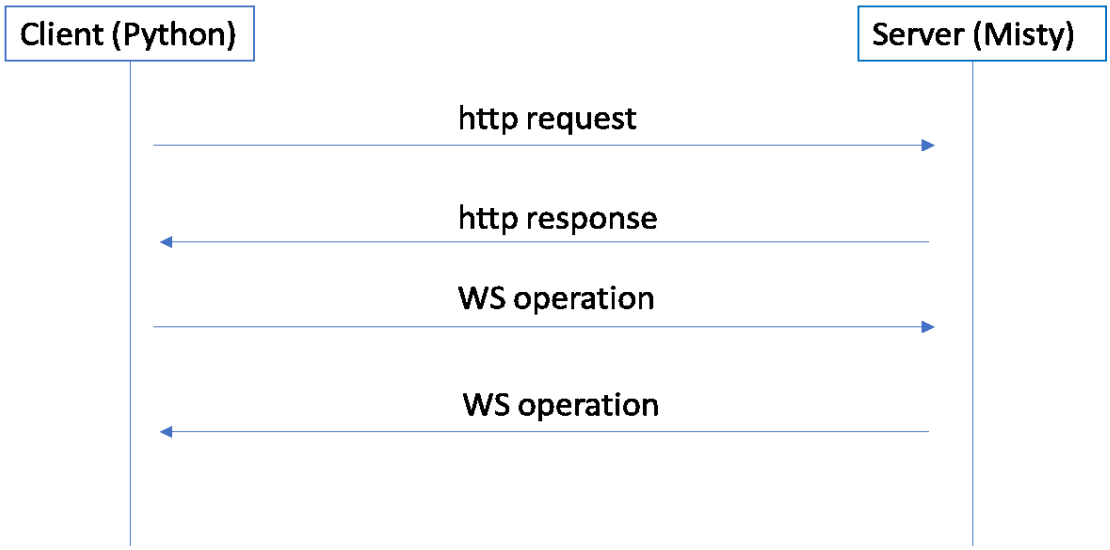


Figure 10: example of WebSocket communication between Robot (Server) and Python (Client).

Figure 8 shows a block scheme of how WebSocket communication between Python and Misty occurs. Python acts as client sending an HTTP request to Misty which responds sending the answer to Python's requests (Algorithm 3).

Algorithm 3 Face Recognition

1. // activation of face recognition skill
2. **input:** http request for face recognition
3. **output:** face recognition values (bearing, distance, elevation, person_name, label ID). Automatic saving of these values into a .txt file.

Robot-assisted Human Indoor Localization

The proposed human indoor localization system is composed of:

1) robot-self localization.

2) human following using face recognition algorithm.

A graphical representation of the proposed localization system is shown in Figure 10. In this section each components of the block scheme in Figure 10 is described end the overall algorithm is presented.

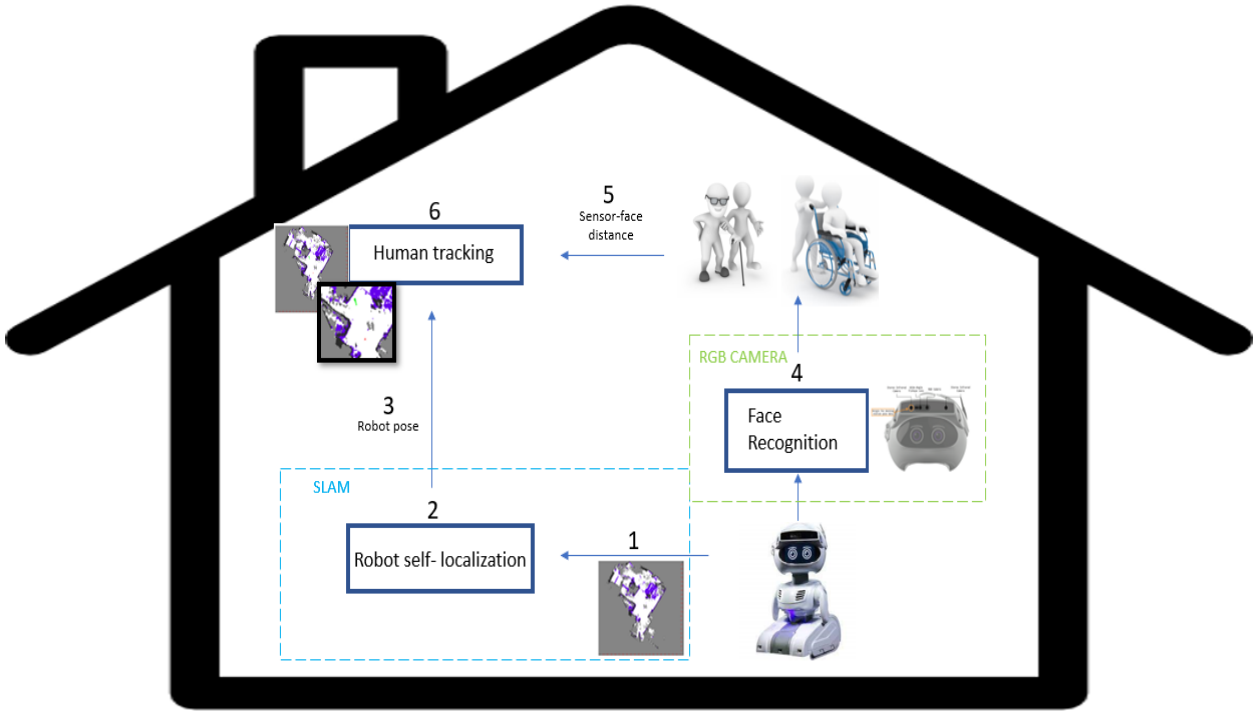


Figure 11: block scheme of algorithm

2.6 Robot Self-Localization

Misty localization and mapping are the first step for the human localization. After map acquisition, map reconstruction and map selection (section 2.3) Misty's pose can be obtained, starting Misty tracking algorithm. In Misty's tracking the Occipital Structure Core depth sensor with stereo infrared cameras are activated, so that Misty can localize itself. The robot self-localization module takes input from sensors and output the robot's current pose (position $\xi_r(k)$ and orientation $\theta_r(k)$). These information are associated with a previously constructed map of the environment.

Algorithm 4

1. // map activation
2. **input:** current active map
3. **output:** map plot ⟸ **Algorithm 1**
4. // activation of tracking function
5. **input:** HTTP request to Misty for tracking activation
6. **output:** Robot position $\xi_r(k)$ and orientation $\theta_r(k)$.
7. **if**

the robot is not able to obtain its pose in 10 seconds it is necessary move it.

2.7 Face Recognition

As described in section 2.5 Face Recognition algorithm is used to recognize the human "Target 1" that you want to follow. Human face recognition algorithm detects selected user and return a list of values, including the distance (d_{r1}) at which the person is detected. The following lines describe briefly which type of data is possible to obtain in output from the algorithm.

Algorithm 5

1. **output: Algorithm 3**
2. //selection of distance of Target 1
3. **input:** values saved on a .txt file
4. **if**
5. person_name= Target 1:
6. convert distance information according to the transformation equation to obtain (d_{r1}).
7. **else**
8. consider the following distance saved on .txt file

In line 6. of the Algorithm 4 the following transformation equation is introduced.

$$\{P_r \text{ X } c_f + [d_s \text{ X } \cos(\theta_h)]\} : c_f \qquad \text{Equation 2.}$$

c_f : conversion factor = meters per cell

P_r : position of robot

d_s : distance returned by the sensor

θ_h : head angle of robot

Equation 2 is necessary because the distance value returned by the face recognition algorithm represents sensor-face distance. A geometric transformation converts this value into coordinates that can be used into pre-bult 2D map of the environment.

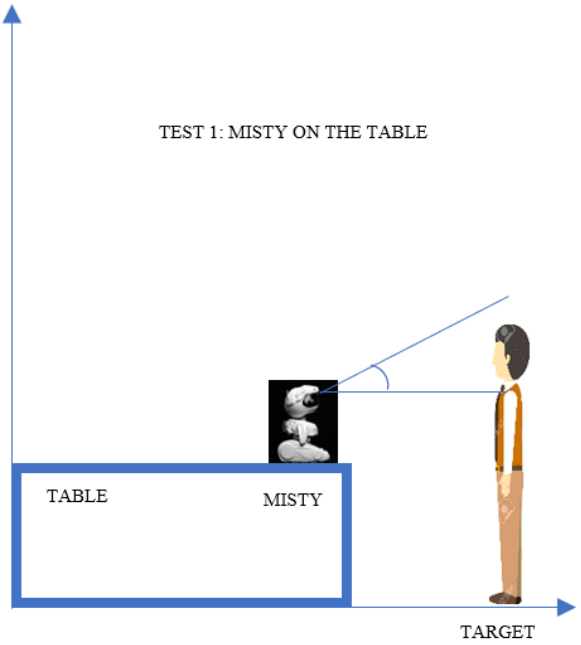


Figure 12: graphic representation of the distance returned by the face recognition algorithm

2.8 Tracking Algorithm

In this section is proposed an algorithm for localization and tracking of Target in internal environment through a combination of the information obtained in the previous sections.

Algorithm 4 Human Indoor Localization Algorithm

1. // map acquisition, map construction and map selection
2. **input:** HTTP requests for last map acquired
3. **output:** reconstructed map ⟸ algorithm 2
4. **input**: HTTP requests for start Misty tracking
5. **output**: position and orientation ⟸ algorithm 4
6. **input**: Target height
7. **output**: best distance for face recognition algorithm
8. **input**: HTTP requests for face recognition
9. **output**: distance of Target 1 ⟸ algorithm 3
10. **if**
11. dr_1 > optimal distance

```
12. input: http request to permit misty movement
13. else:
14. plot of distance on map
```

The algorithm uses the 2D reconstructed map, Misty's pose, and the facial recognition algorithm to reconstruct the target's coordinates on the map. Iterating this process is possible to pass from user detection to user tracking. The following sections illustrate the main issues that have been encountered and how these have been overcome to obtain a more accurate algorithm for human localization.

The first issue is the user distance reconstruction on the acquired map. As mentioned before, the user coordinates obtained as output from Algorithm 2 must be drawn up on the environmental map to perform human detection.

Misty’s map position appears like a point so user’s coordinates, obtained converting distance output of face recognition algorithm, could be any point on circle centred on Misty (Figure 12). To reduce this information to a single point on this circumference is used Misty orientation information obtained from its pose. Misty reference system orientation presents the same orientation of map environment as showed in Figure 12. Misty orientation angle gives information on how this reference frame is oriented respect to the map one. Using this information is possible to know how Misty head is orientated and so the direction in which face detection is performed.

Another problem I faced, was the angle of rotation of Misty's head. Thinking that Misty can be placed in different indoor environments and that the people on whom robot performs the facial recognition have a very different height, it is important to know what is the distance at which the facial recognition algorithm can detect the user and the optimal Misty head angle in relation to the user’s height. This information is used to modify the algorithm to move Misty away or close to the target while increasing accuracy.

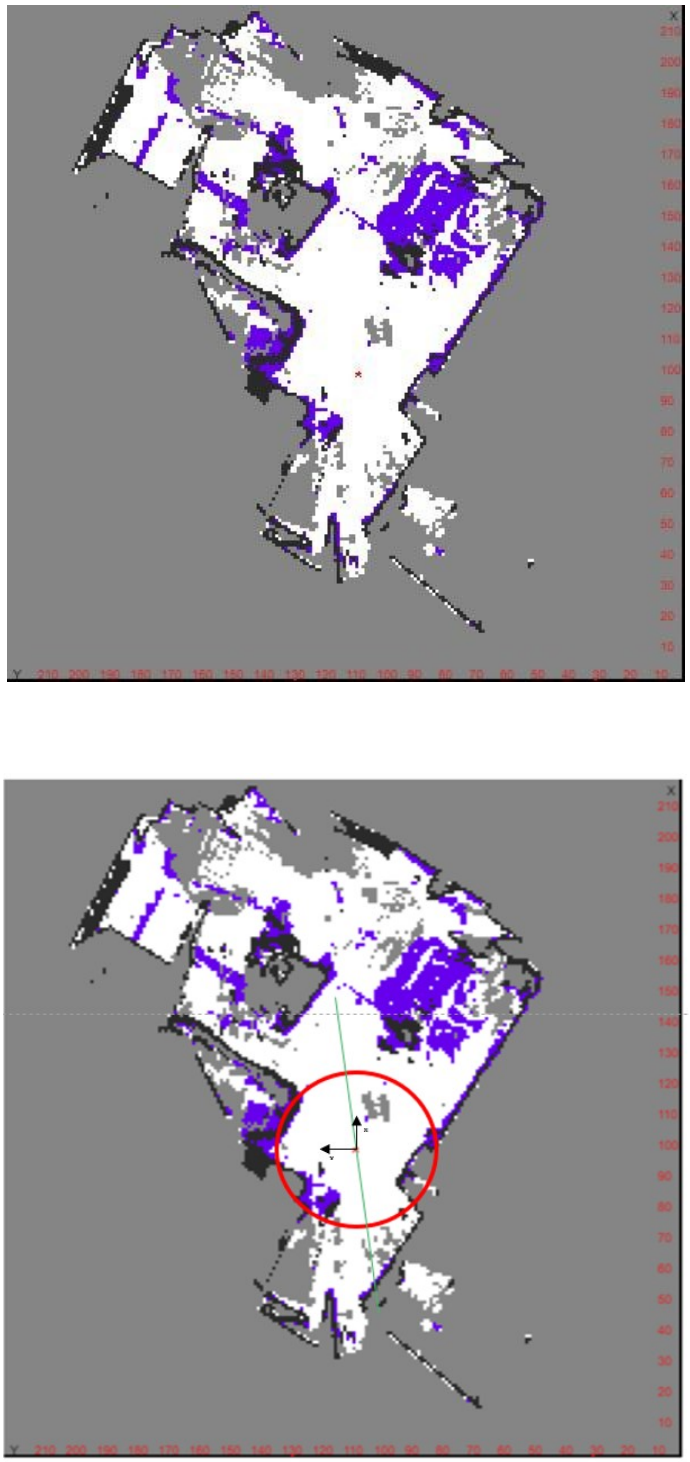


Figure 13: on top: reconstructed robot map. On bottom: reconstructed robot map that includes robot position (red star), possible area where the target can be located (red circle) and computed direction rotation of robot head (green line). The robot reference frame is expressed as two black arrows plotted on robot position on the map.

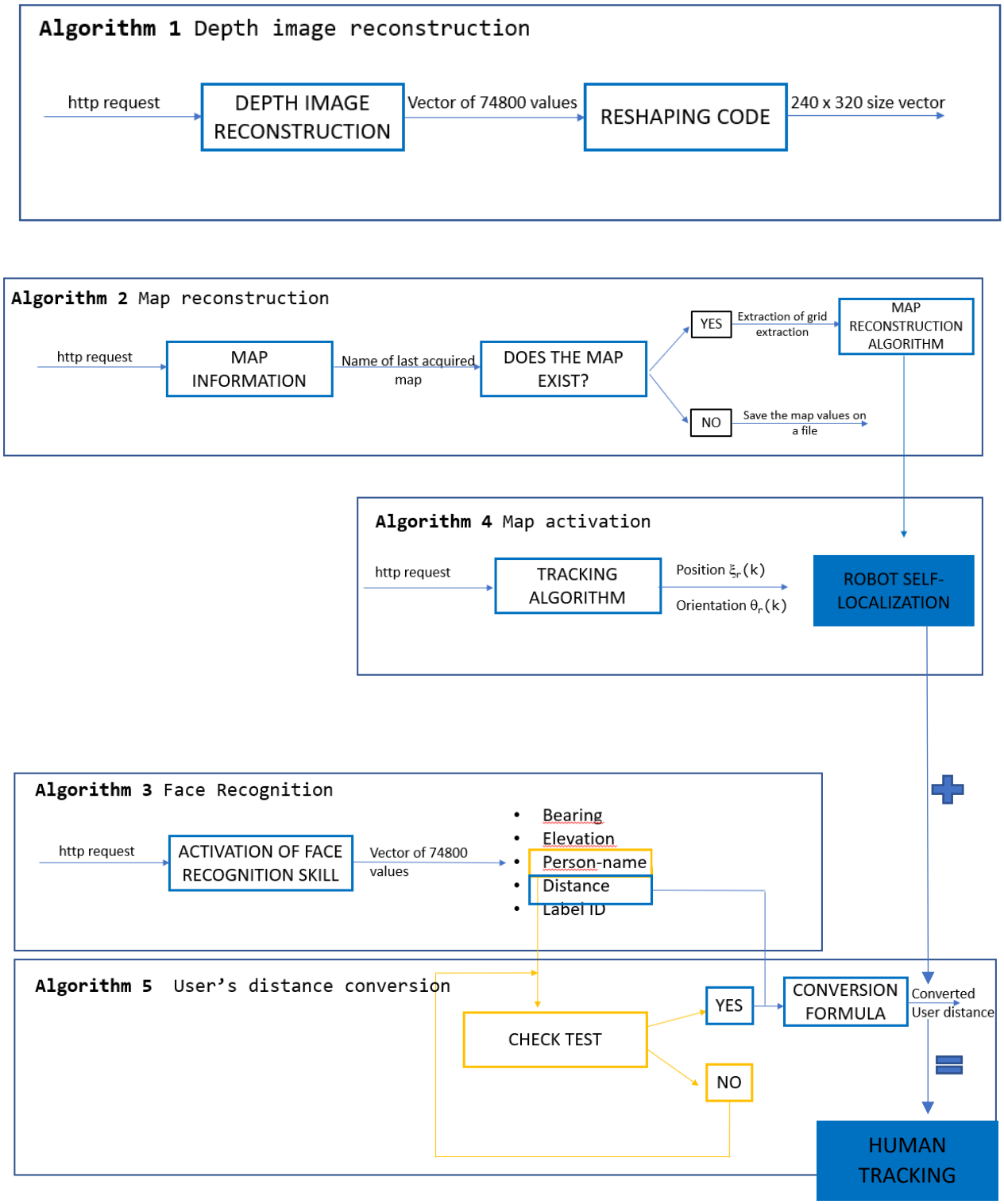


Figure 14: Block diagram of the algorithms used. For each algorithm, the input and output of each single function used is highlighted with an arrow.
The vertical arrows indicate the connection between the output of a function of an algorithm and the function of the other algorithm to which it is connected. The functions inside the blue blocks are the main functions shown in the figure 10.

2.9 Experimental setup

In this thesis human tracking and localization is performed using Face Recognition algorithm. To be sure that this algorithm gives acceptable values of distance, an experimental validation is performed. The experiments were conducted in a 5.5 by 6.5 square meters [m^2] lab area of UNIVPM (Università Politecnica delle Marche) as shown in Figure 14. The algorithm run on an ASUS laptop and the robot was set in two configurations. In the first configuration the robot was located on a table of 75 cm height. The user stand in front of Misty at six different distances marked on floor (Figure 15) with different Misty's head degrees (10°, 20°, 30°). The user was asked to move to the next predetermined distance only when the face recognition algorithm was stopped. The first half of user's tests was performed in the morning when sun's ray no penetrating the room, while the second half was performed in the afternoon when a big amount of sun's ray penetrating the room and Misty's face recognition sensor was hit by them.

To have a big dataset for the analysis, eleven users with heights ranged between 165 cm and 195 were selected.

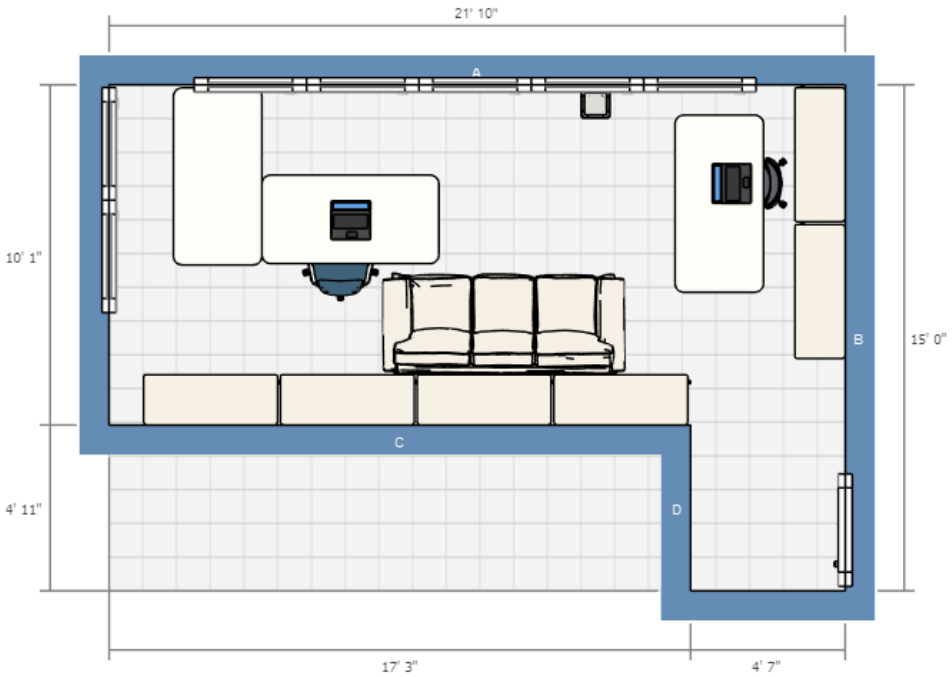


Figure 15: Reconstruction of the laboratory chosen to carry out the experiments

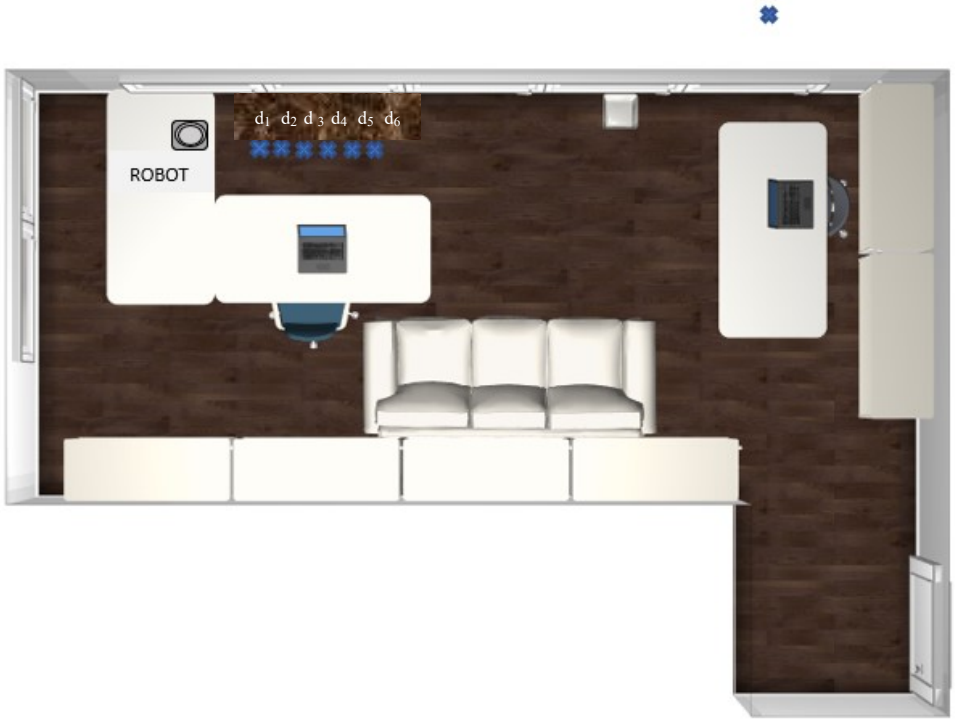


Figure 16: First setup configuration. The robot is located on the table. The blue crosses indicate the positions at which the target is located during the experiments.

Figure 17: Second setup configuration. The robot is located on the floor. The blue crosses indicate the positions at which the target is located during the experiments.

In the second configuration the robot was located on the floor (Figure 16). The user was asked to locate themselves on same makers on the floor of the configuration number one. Three repetitions of the experiment for each subject ware performed with different Misty's head angle. For this configuration Misty's head degrees was set to 20°, 30°, 40° increasing the probability to capture faces at small distances from robot. Ones the validation tests were performed and the errors on face recognition distances are computed a criterion for targets classification dependently from distance and from robot head configuration is established. The presence of a trend among target's height and measurement uncertainty are also evaluated. After these procedures, the entire Localization and Tracking algorithm is tested.

For the tests on this algorithm to the Targets was asked to perform firstly a linear path in front of Misty (Figure 17) and then to them was asked to follow a nonlinear path (Figure 18). Each Target repeats both paths one time for each of the two Misty configurations.

Figure 18: First test performed: the robot is located first on the table with three head angles and then on the floor with three head angles. To the subject is asked to follow a linear path (blue line). The blue star indicates the starting point of the linear path followed by the user, while with the blue circle indicates the stopping point.

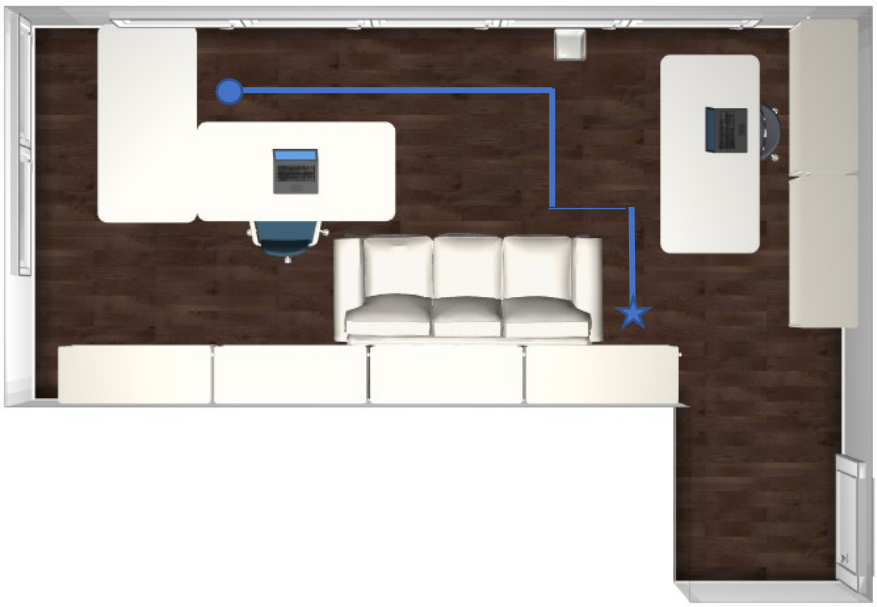


Figure 19: second test performed: the robot is located first on the table with three head angles and then on the floor with three head angles. To the subject is asked to follow a non-linear path (blue line). The blue star indicates the starting point of the linear path followed by the user, while with the blue circle indicates the stopping point.

Instead for the Face Recognition algorithm the repeatability tests and the global uncertainty tests are performed.

The repeatability tests are performed to establish the uncertainty measurement on fixed distances for each studied robot head angle independently from human height. The global uncertainty instead expresses the frequency with which the error associated to the difference between measured and real data occurs. For this measurement, the statistical confidence is computed.

For the computation of the input-output relationship the calibration curve was used. To find the line that best interpolates the dispersed points, the method of least squares was chosen. The equation considered for the straight line is:

$$q_o = mq_i + b \qquad \text{Equation 3}$$

where

q_o= output (independent variable)

q_i= input (independent variable)

m = angular coefficient of the straight line

b = point of intersection between the straight line and the vertical axis (intercept at the origin)

The equations used to compute m and b are:

$$m = \frac{N\sum q_i q_o - (\sum q_i)(\sum q_o)}{N\sum q_i{}^2 - (\sum q_i)^2} \qquad \text{Equation 4}$$

$$b = \frac{(\sum q_o)(\sum q_i\hat{}2) - (\sum q_i q_o)(\sum q_i)}{N\sum q_i{}^2 - (\sum q_i)^2} \qquad \text{Equation 5}$$

N= total number of observations

The standard deviation of estimated values for *m* and b can be found as:

$$s^2{}_m = \frac{N s^2{}_{qo} \sum q_i{}^2}{N\sum q_i{}^2 - (\sum q_i)^2} \qquad \text{Equation 6}$$

$$s^2{}_b = \frac{s^2{}_{qo}}{N\sum q_i{}^2 - (\sum q_i)^2} \qquad \text{Equation 7}$$

$$s^2{}_{qo} = \frac{1}{N-2}\sum(m\,q_i + B - q_o)2 \qquad \text{Equation 8}$$

Where s_{qo} represents the standard deviation of q_o. This means that if q_i were fixed and then repeated several times, q_o would return data with a certain dispersion. The standard deviation of the input is computed starting from the standard deviation of the output and the value of m^2 .

Chapter 3

Results

3. Results

After the algorithms were tested, the data acquisition was performed. These data are managed in order to extract some useful information for the correction of the localization and tracking algorithm all the reorganized data will be commented in the next sections.

3.1 Results of the human tracking

In section 2.2.1 an experimental protocol for the optimal distance at which the person can be detected using a photo capture by the 4K camera and a YOLO algorithm is presented. The algorithm results are showed in Table 4. For the pitch angle of robot head three configuration are chosen. Small angles are ranged between 0° and 5°, medium angles are ranged between 6° and 20° and high angles are ranged between 21° and 29°. The angles values are expressed as absolute values.

Table 4: results of experimental protocol for testing robot vision capability

DISTANCE FROM HUMAN TO ROBOT				PITCH ANGLE OF ROBOT HEAD			ENVIRONMENTAL CONDITION			HUMAN SIZE		ACCURACY		HUMAN DETECTION	
0 m-1m	1 m-2m	2m-3m	>3 m	Small	Medium	High	object	room size	n.of peaple					YES	NO
x				x			no		1	1.78					x
	x			x			no		1	1.78					x
		x		x			no		1	1.78				x	
x						x	no		1	1.78					x
	x					x	no		1	1.78				x	
		x				x	no		1	1.78				x	
			x			x	no		1	1.78				x	
x					x		no		1	1.78					x
	x				x		no		1	1.78				x	
		x			x		no		1	1.78				x	
			x		x		no		1	1.78				x	

The results suggest that the best condition in which is possible to detect a person using the 4K camera is when the distance human-robot is at least of 1 meter and the pitch angle of robot head is at least of 6°. The different scenarios tested are showed in the next figure (Figure 20, Figure 21, Figure 22), in which a reproduction of normal office condition is presented.

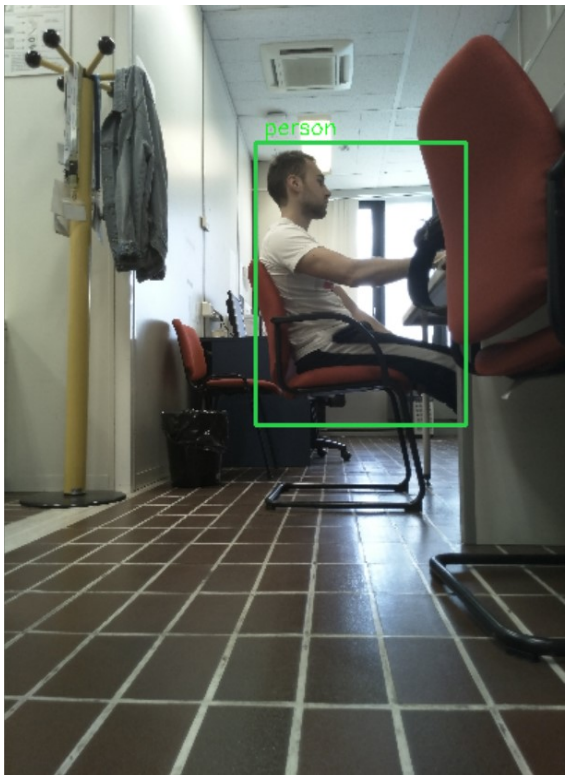


Figure 20: validation of YOLO algorithm on single subject

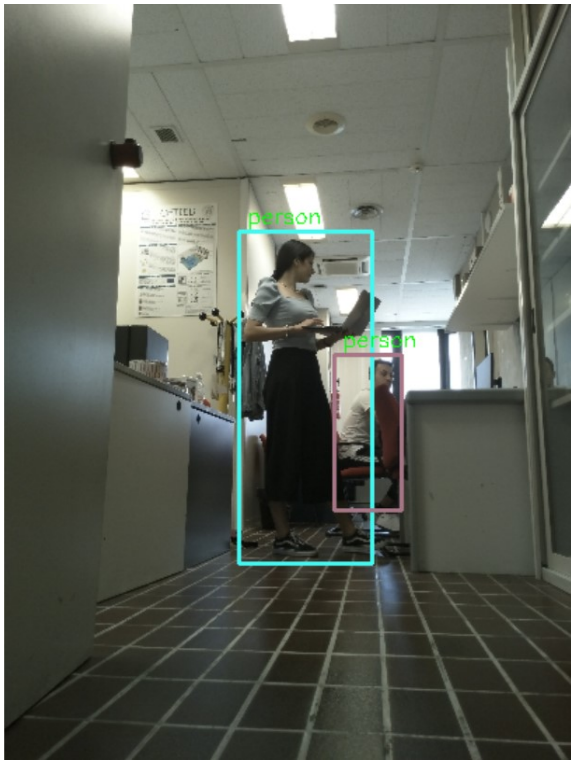


Figure 21: validation of YOLO algorithm on image with multiple subjects

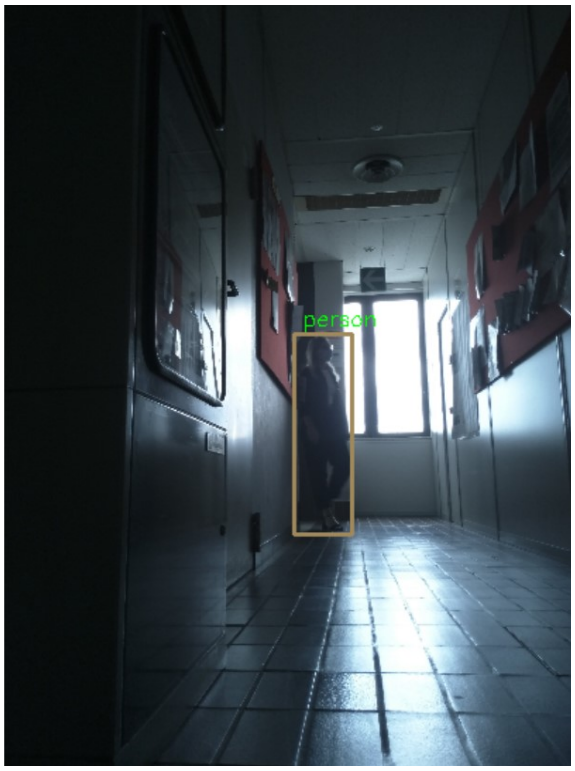


Figure 22: validation of YOLO algorithm on an image when the subject is in the shade

As visible in the pictures the results suggest that the modified YOLO algorithm is a very efficient algorithm capable to detect person when they are partially occluded, when more than one subject is present in the scene and under bad light condition like showed in figure 22 in which the person is completely in darkness. With these data is possible to understand how the robot see the world using the RGB camera and how the light is not an impediment to the vision of the Target. In addition, this protocol reveals to be an important starting point when Face Recognition algorithm was performed because gives information on Misty's head angle at which is possible to see the Face of a person.

Unfortunately, this solution cannot be accepted for our purpose because as reported in state of art this method requires the acquisition of images causing a lack of privacy police.

3.2 Experimental Results of Face Recognition algorithm

In this subchapter it will be presented the experimental results related to the validation of Face Recognition algorithm.

Firstly, the results obtained for the repeatability tests will be presented. Then we focused on the results obtained to compute the global uncertainty. In addition to these two tests

one more data classification is computed to understand if a relation among user' heights and measurement errors exists. For each of these tests results related to robot located on the floor and on the table are reported.

3.2.1 Repeatability test

For the repeatability test data (q_i) the fixed distances at which each subject performed the test are considered as input. The output data (q_o) are represented by the standard deviation of the residues (σ) which are the difference between input and output data (Equation 4). The results of this test are user independent.

$$q_{o=}\, std(q_i - q_o) \qquad \text{Equation 9}$$

3.2.1. A. Repeatability test results for robot located on table

The data acquired when the robot is located on the table, were classified based on fixed distances. Appling the Equation 9, the standard deviation of residues (σ) is computed. The relationship between distances (input data) and output (residues standard deviation) is showed in two different ways in Figure 23 and in Figure 24.

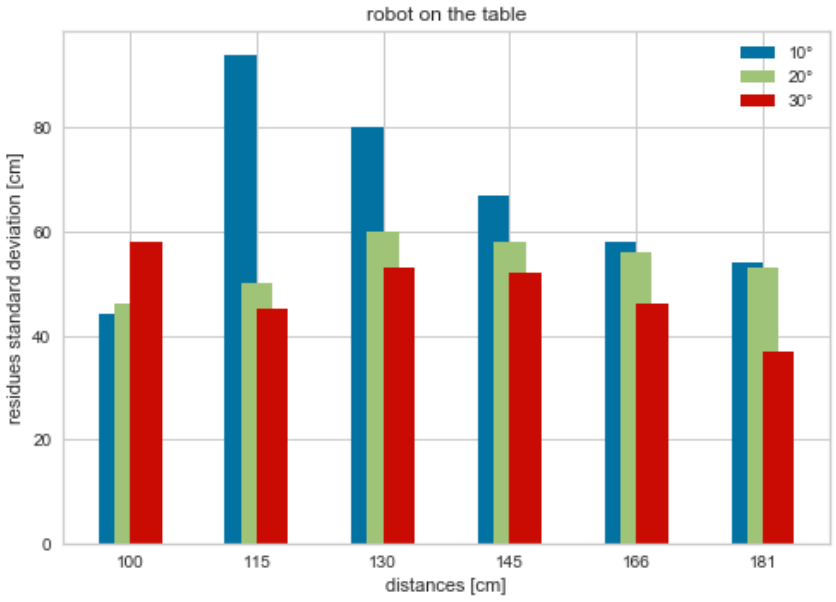


Figure 23: histogram showing how the standard deviation of the residuals varies in the fixed distances. The results for the angle of the robot head equal to 10° are shown in blue. The results for the angle of the robot head equal to 20° are shown in green. the results for the angle of the robot head equal to 30° are shown in red.

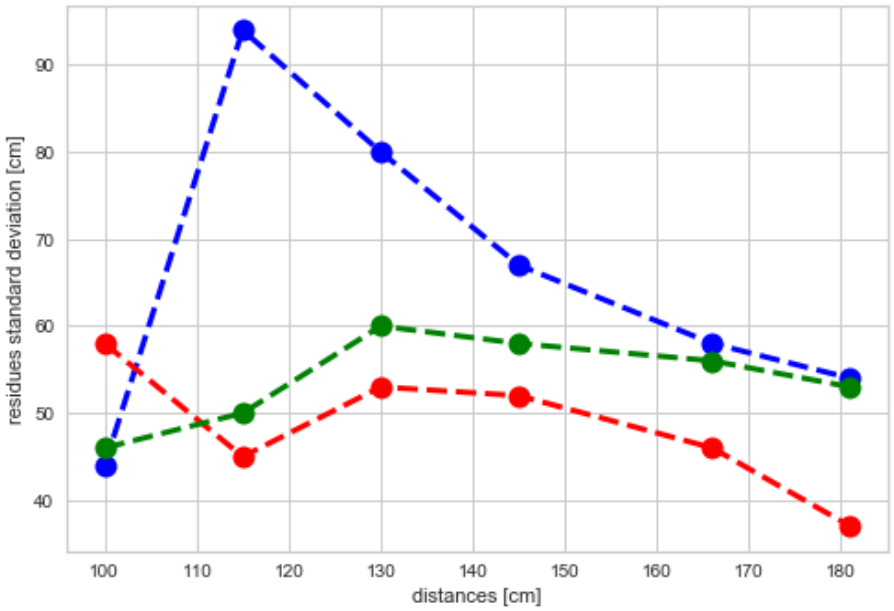


Figure 24: graph representing the exact trend of the variation of the standard deviation of the residuals with the distances. for the angle of the robot head equal to 10° are shown in blue. The results for the angle of the robot head equal to 20° are shown in green. the results for the angle of the robot head equal to 30° are shown in red.

In these figures it is possible to see how the standard deviation of the residues for a robot head angle at 10° and at 20° firstly increases until a maximus is reached and then decreases with the increase of fixed distances. Contrary, standard deviation of the residues for a robot head angle of 30° firstly decreases until a minimum is reached, then increases while remaining below the maximum value and at the end decreases.

The Table 5 reports the single residues standard deviation values computed for each fixed distance. The mean value of each robot head angle is also reported.

Table 5: Values used to compute the standard deviation of the residues for each robot head angle. In addition, a mean of these value is calculated. The sign ± indicates the range of variation of the standard deviation of the residues.

ROBOT ON TABLE	± 2σ						
FIX DISTANCES [cm] / ROBOT HEAD ANGLE	**100**	**115**	**130**	**145**	**166**	**181**	**MEAN**
10°	±44	±94	±80	±67	±58	±54	66
20°	±46	±50	±60	±58	±56	±53	53
30°	±58	±45	±53	±52	±46	±37	48
MEAN	49.3	63	64	59	53.3	48	

3.2.1. B. Repeatability test results for robot located on floor

In the second repeatability test the data that has been used are those acquired when the robot is placed on the floor. Also in this case a classification based on fixed distances was done. Appling the Equation 9, the standard deviation of residues (σ) is computed. The relationship between distances (input data) and output (residues standard deviation) is showed in two different ways in Figure 25 and in Figure 26.

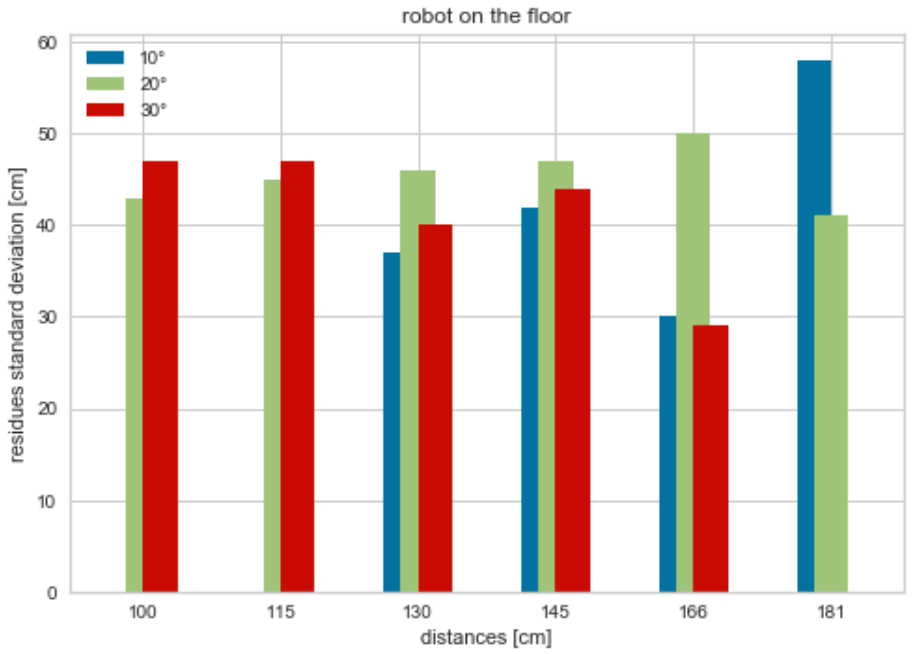


Figure 25: histogram showing how the standard deviation of the residuals varies in the fixed distances. The results for the angle of the robot head equal to 10° are shown in blue. The results for the angle of the robot head equal to 20° are shown in green. the results for the angle of the robot head equal to 30° are shown in red.

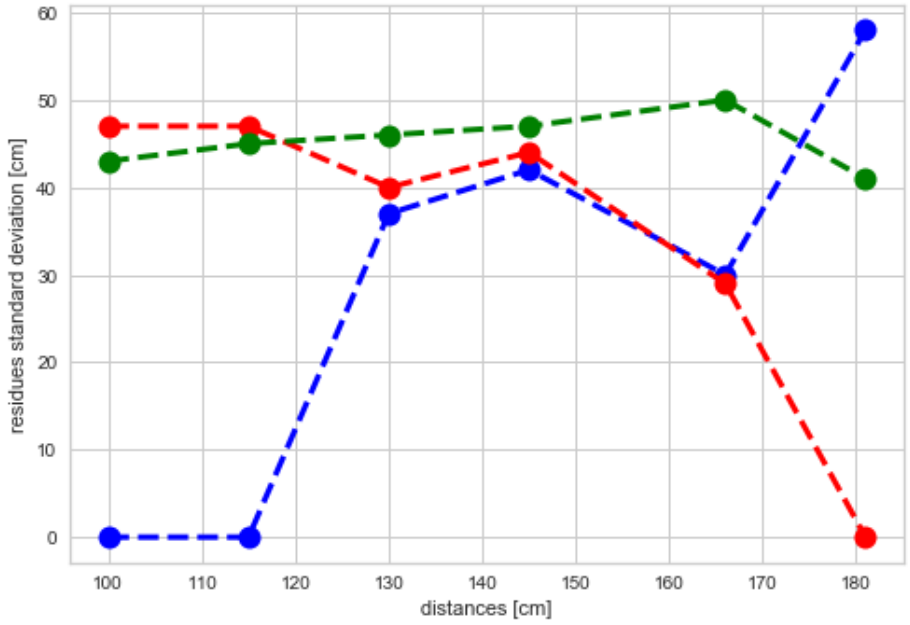


Figure 26: graph representing the exact trend of the variation of the standard deviation of the residuals with the distances. for the angle of the robot head equal to 10° are shown in blue. The results for the angle of the robot head equal to 20° are shown in green. the results for the angle of the robot head equal to 30° are shown in red.

From figures is possible to note that for 100 cm and 115 cm distances there are no records of data for 20° tilt of robot head and that for 181 cm distance there is no record for 40° 20° tilt of robot head. In these figures is possible to see how the standard deviation of the residues for a robot head angle of 20° starts to be calculated for a distance of 130 cm because for shorter distances the Face Recognition algorithm was not able to capture faces. The standard deviation of residues trend first increase, then decrease for increasing again until a maximum is reached.
A trend that grows linearly as the distance increases is observable if an analysis of the curve representing the standard deviation of the residuals for 30 ° angle of the robot head is done. For the last distance, the standard deviation of the residuals tends to stabilize around the starting value. standard deviation of the residues for a robot head angle of 40° firstly decreases, then increases while remaining below the maximum value and at the end decreases. In the last distance no value was recorded so the curve goes to zero. Table 6 reports the single residues standard deviation values computed for each fixed distance. The mean value of each robot head angle is also reported. The empty spaces represent the distances for the angle at which Face Recognition algorithm is not able to detect Targets. In Figure 27 an overview of the results founded for the repeatability test of the robot configurations are showed.

Table 6: figure showing the exact values of standard deviation used to compute the standard deviation of the residues for each robot head angle. In addition, a mean of these value is calculated. The sign ± indicates the range of variation of the standard deviation of the residues.

ROBOT ON FLOOR	± 2σ						
FIX DISTANCES [cm] / ROBOT HEAD ANGLE	100	115	130	145	166	181	MEAN
20°			±37	±42	±30	±58	18
30°	±43	±45	±46	±47	±50	±41	45
40°	±47	±47	±39	±44	±29		34
MEAN	30	30.6	40.6	44.3	36.3	33	43

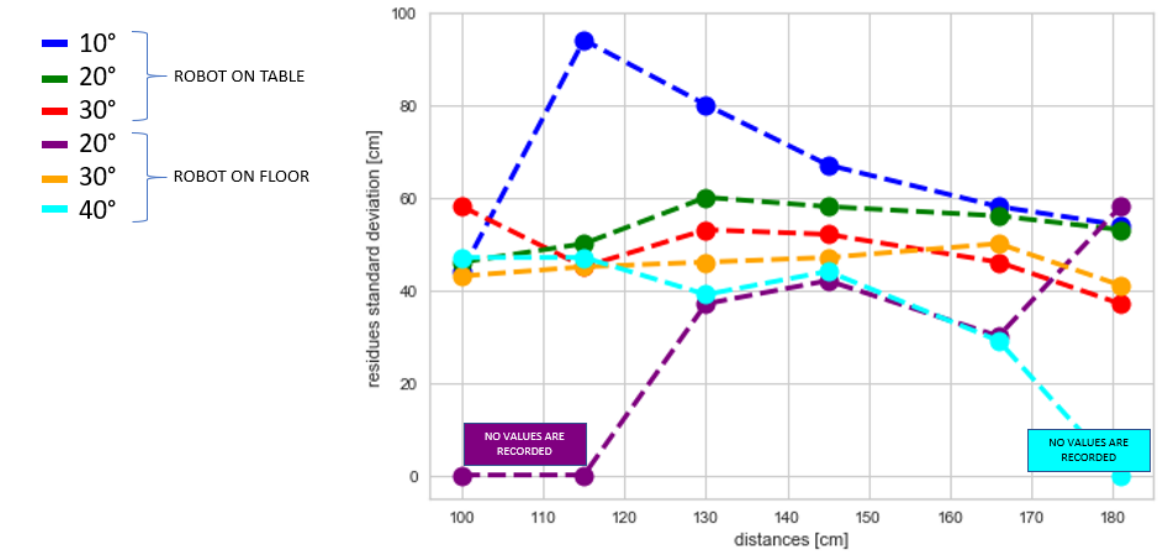


Figure 27: figure showing a global trend of the repeatability tests for all the chosen angle and for all the two robot head configuration. The value that goes to zero represent the value for Which Face Recognition algorithm does not provide results.

3.2.2 Global uncertainty test

For the global uncertainty test a merge of all the subjects independently from the fixed distances is performed. This test suggests what is the probability for a certain residues value to occur for each of the chosen robot head angle. Additionally, a statistical confidence, expresses as a percentage and with a covering factor equal to two, is computed. The results are reported for the two built setups.

3.2.2. A. Global uncertainty results for robot located on table

Figure 28, Figure 29, and Figure 30 represent the results obtained from the global uncertainty test for robot located on table. It is possible to observe how for robot head angle of 10° the algorithm tends to overestimate the input data. For robot head angle of 20° the algorithm overestimates the input data. A partial condition of equilibrium occurs for robot head angle of 30°.

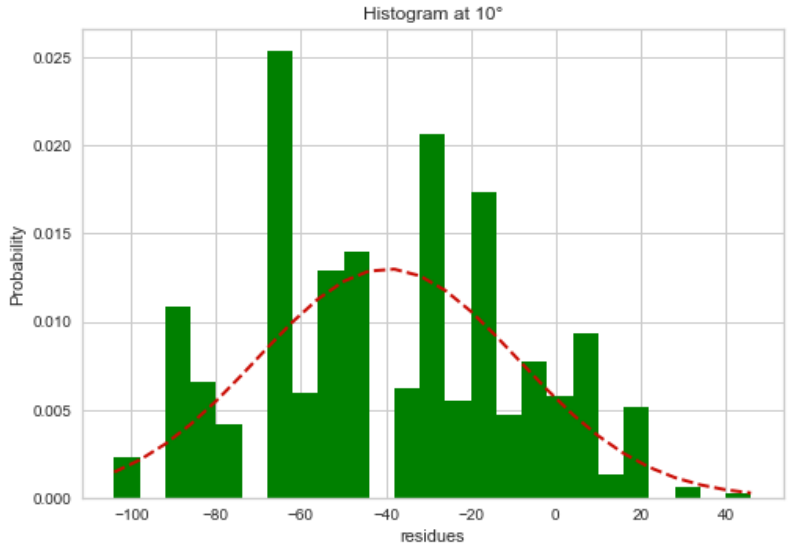


Figure 28: histogram representing the frequency of residues for robot head angle equal to 10°. The red line represents the normal distribution of the residues.

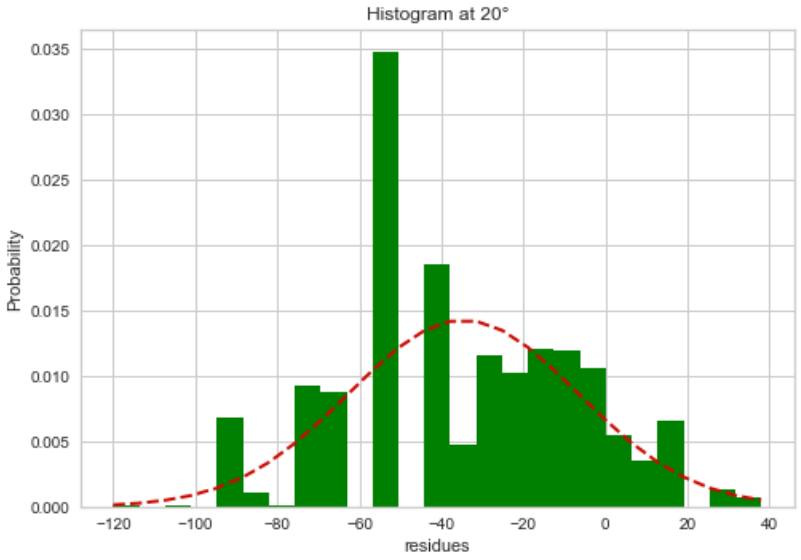


Figure 29: histogram representing the frequency of residues for robot head angle equal to 20°. The red line represents the normal distribution of the residues

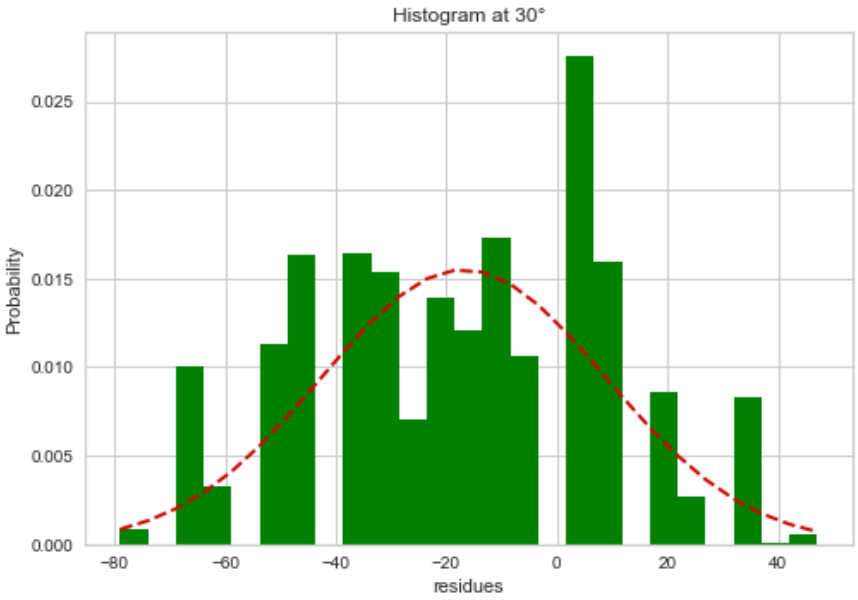


Figure 30: histogram representing the frequency of residues for robot head angle equal to 30°. The red line represents the normal distribution of the residues

Table 7: table representing the values of statistical confidence for a coverage factor equal to one (K=1) and two (K=2) for each evaluated robot head angle

ROBOT ON TABLE	STATISTICAL CONFIDENCE	
	K=1	**K=2**
10°	29.21 %	58.42 %
20°	25.58 %	51.02 %
30°	28.1 %	56.2%

In Table 7 the values of statistical confidence are reported for a coverage factor equal to one and equal to two. As previously said the result considers in this thesis is the one for statistical confidence with a coverage factor equal to two.

3.2.2. B. Global uncertainty results for robot located on floor

The Figure 31, Figure 32 and Figure 33 represent the results obtained from the global uncertainty test for robot located on floor. It is possible to observe how for robot head angle of 20° the algorithm neither overestimates neither underestimates the input data, while for robot head angle of 30° the algorithm overestimates the input data. A similar condition occurs for robot head angle of 40°.

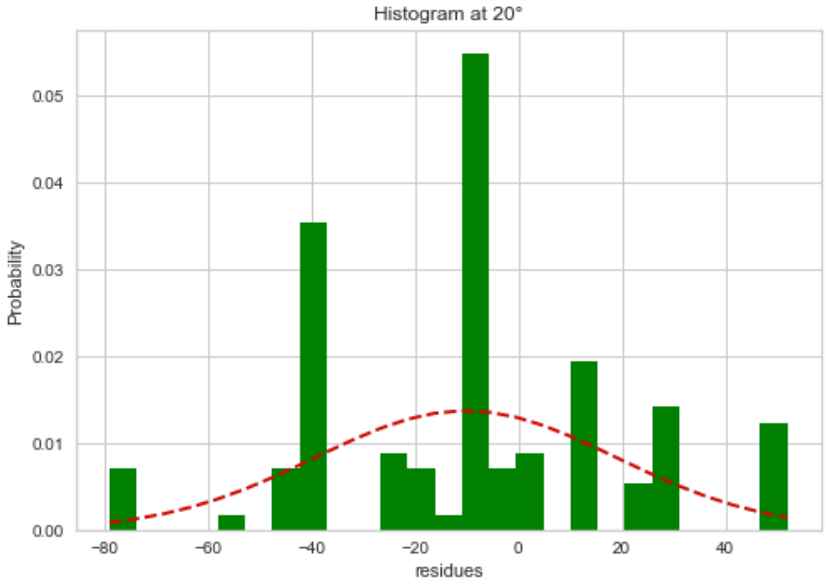


Figure 31: histogram representing the frequency of residues for robot head angle equal to 20°. The red line represents the normal distribution of the residues

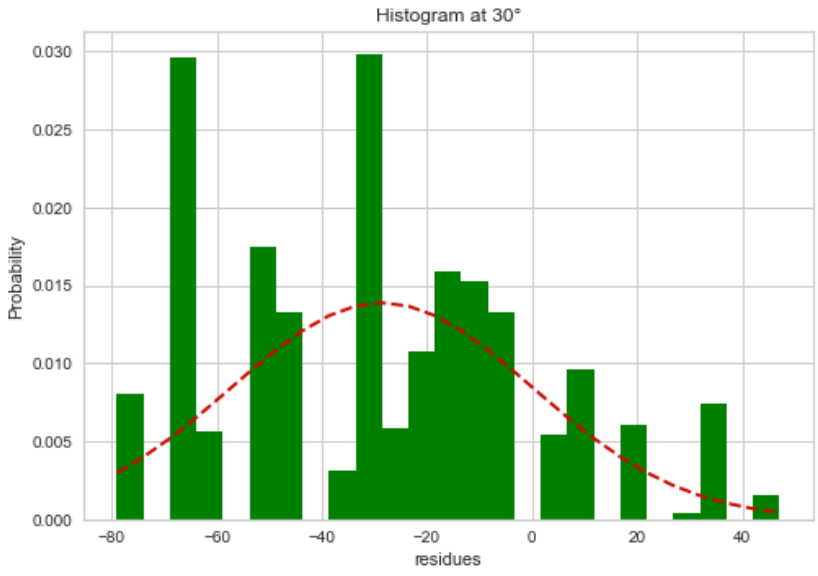


Figure 32: histogram representing the frequency of residues for robot head angle equal to 30°. The red line represents the normal distribution of the residues.

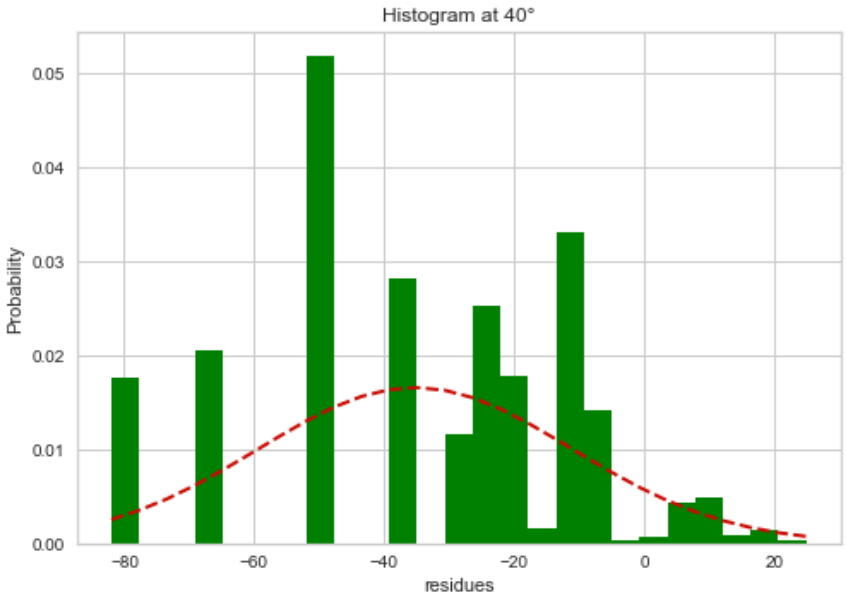


Figure 33: histogram representing the frequency of residues for robot head angle equal to 40°. The red line represents the normal distribution of the residues.

Table 8: table representing the values of statistical confidence for a coverage factor equal to one (K=1) and two (K=2) for each evaluated robot head angle.

ROBOT ON FLOOR	STATISTICAL CONFIDENCE	
	K=1	**K=2**
20°	46.08 %	92.16%
30°	22.55 %	45.1 %
40°	60.84%	121.68 %

In Table 8 the values of statistical confidence are reported for a coverage factor equal to one and equal to two. Also for this setup the result considers in this thesis is the one for statistical confidence with a coverage factor equal to two.

3.2.3 Classification 1

To understand if a certain relationship exists among subject heights and the standard deviation of the input, another classification of the data is performed. The subjects are classified according to increasing height values and for each of them a standard deviation of the input for each robot's head angle is calculated. The results are showed

in a histogram (Figure 35) and as single points to better have an idea of their trend (Figure 34). As visible in these figures seems to be no relationship among height of the subject and standard deviation of the input data for all the three-robot head angle studied.

For the second setup a graphical representation of the results was not possible due to the lack of data acquisition during the face Recognition.

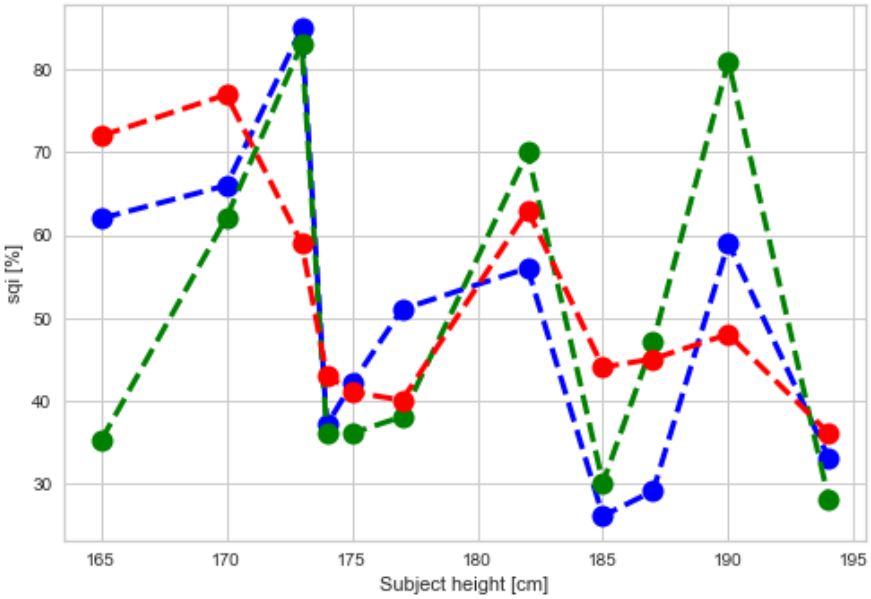


Figure 34:single point representation of the results obtained analysing the standard deviation of the input value for each subject. The subjects are classified according to increasing heights

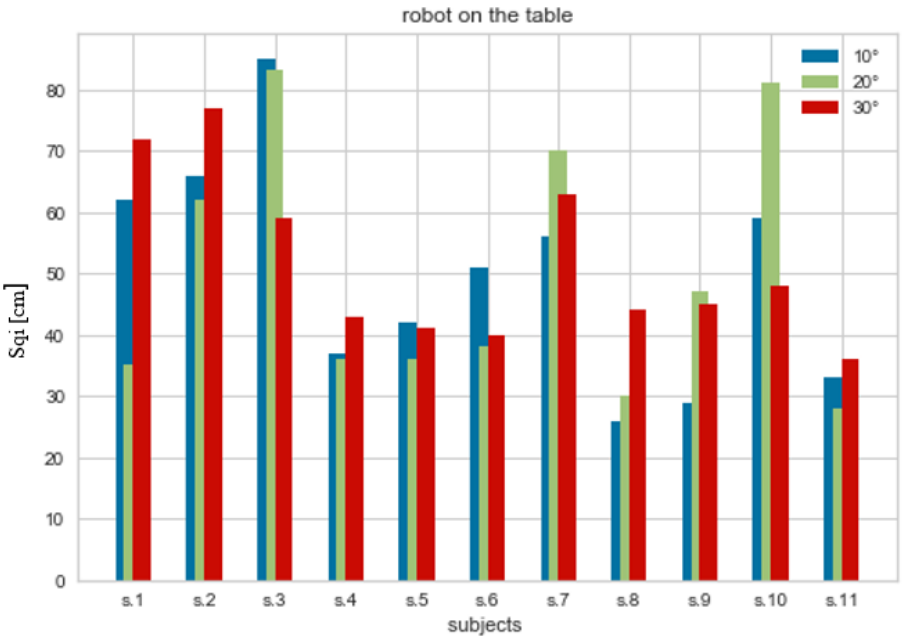


Figure 35: histogram of the results obtained analysing the standard deviation of the input value for each subject. The subjects are classified according to increasing heights

3.2.4 Classification 2

A classification based on range of heights is performed. From the classification seven groups for eleven Targets heights are extracted. The classification is performed considering ranges of 5 cm. The results related to the first robot setup, i.e. when the robot is located on the table are reported in figure 36 and figure 37. Also, in this case seems that a correlation among subject height and standard deviation of the input does no exists. As in the previous condition two representation of the results are reported for each robot setup.

For the second setup a graphical representation of the results was not possible due to the lack of data acquisition during the face Recognition.

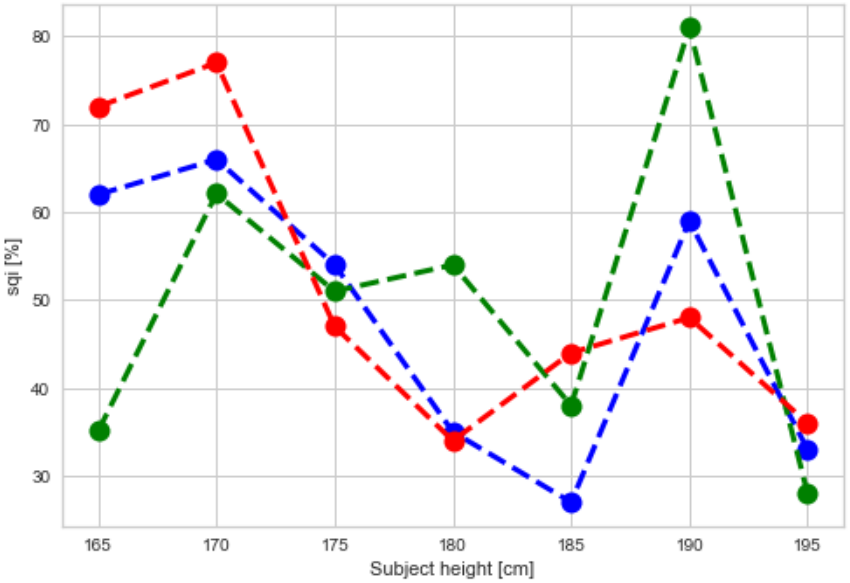


Figure 36: single point representation of the results obtained analysing the standard deviation of the input value for each subject. The subjects are classified according to heights ranges

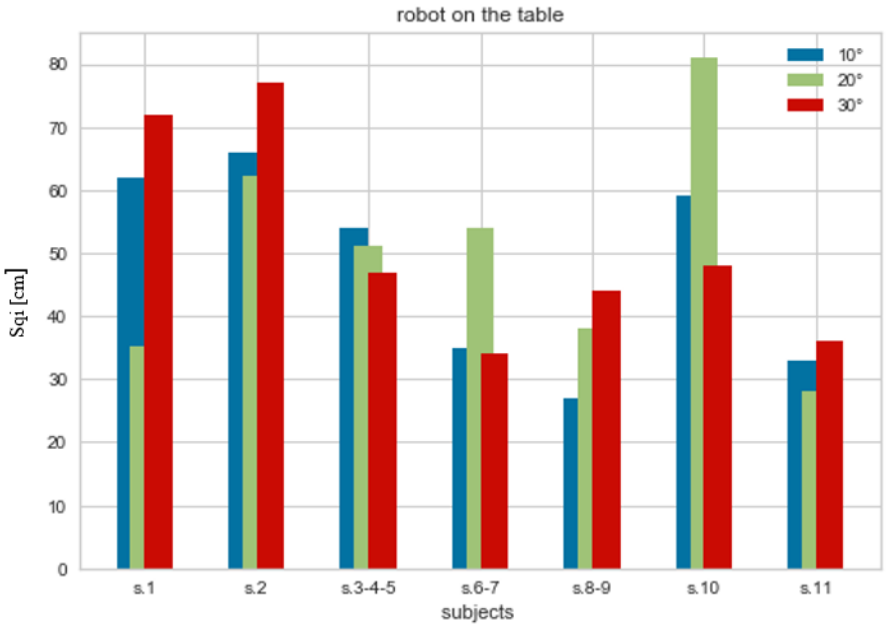


Figure 37: histogram of the results obtained analysing the standard deviation of the input value for each subject. The subjects are classified according to heights ranges

3.2.5 Localization and Tracking Algorithm

After the validation of the Face Recognition algorithm, the entire Localization and Tracking algorithm (Algorithm 6 described in section 2.8) is tested to see if it is

effectively possible to track a human target in indoor environment, using Misty's onboard sensors coupled with a dedicated algorithm.

As described in Section 2.9 to the target was asked to follow a linear path, along a straight line, and a non-linear one, while the robot is placed on the table and when the robot is on the floor.

Figure 38 shows the tracking of the user performed when the robot is located on the table with a robot head of 10°, and the Target follows a path that is linear (panel A) and nonlinear (panel B). The dushed purple line represents the real path that the Target follows, while the dushed green line represents the tracking performed by the algorithm. In Figure 39 it is showed the tracking of the user performed when the robot head has an angle of 20° and when the Target follow a linear path (panel A) and a non- linear path (panel B). Also in this case the two dashed lines represent respectively the real path (purple) follow by the user and the path coming out from the algorithm (green). The Figure 40 shows the algorithm result when the robot head is at 30°. The colours of the legend have the same meaning of the two previous cases.

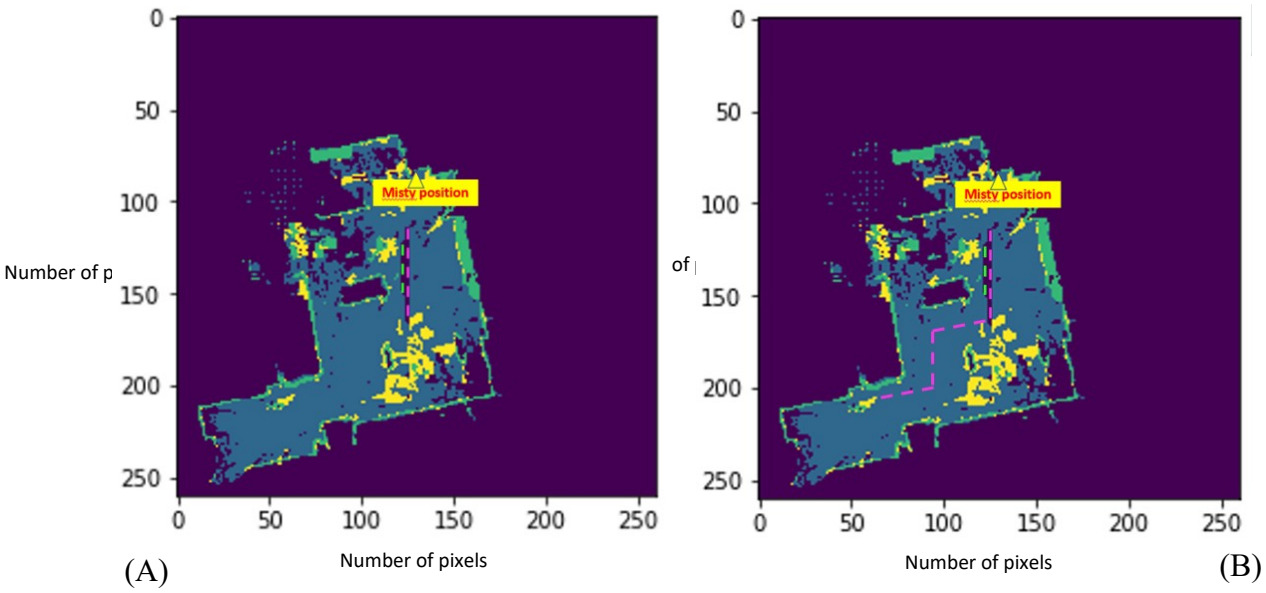
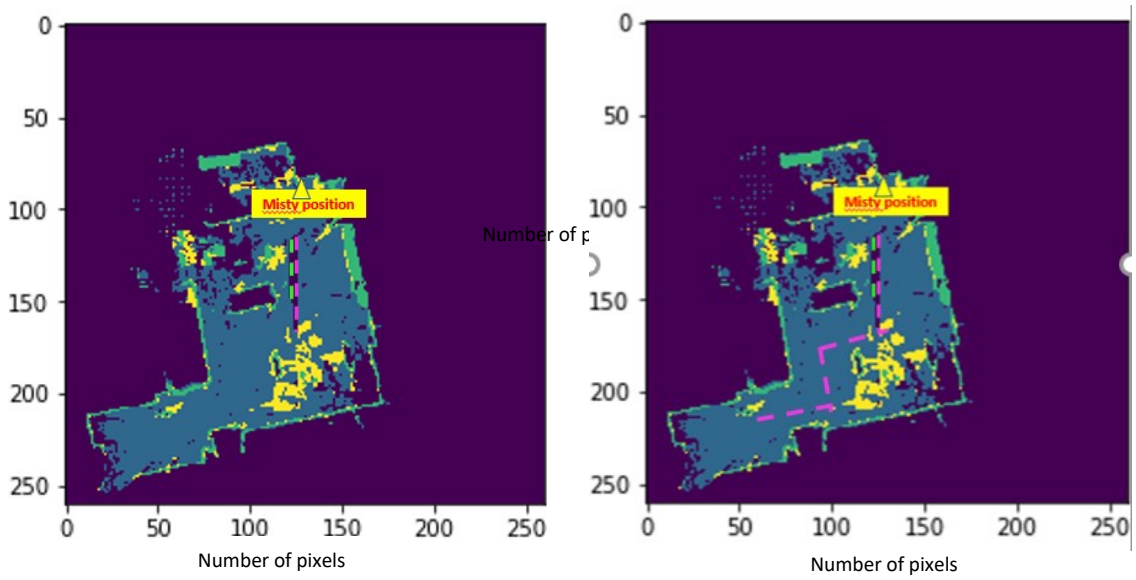
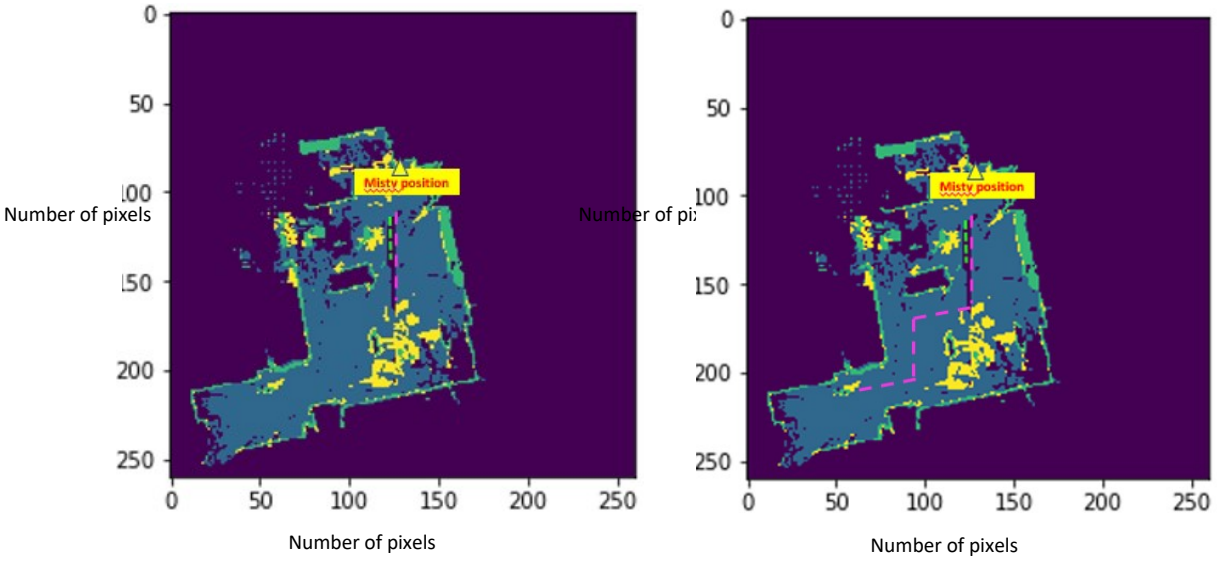


Figure 38: panel A: the dushed purple line shows the linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm. Panel B: the dushed purple line shows the non-linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm.



(A) (B)

Figure 39: panel A: the dushed purple line shows the linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm. Panel B: the dushed purple line shows the non-linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm

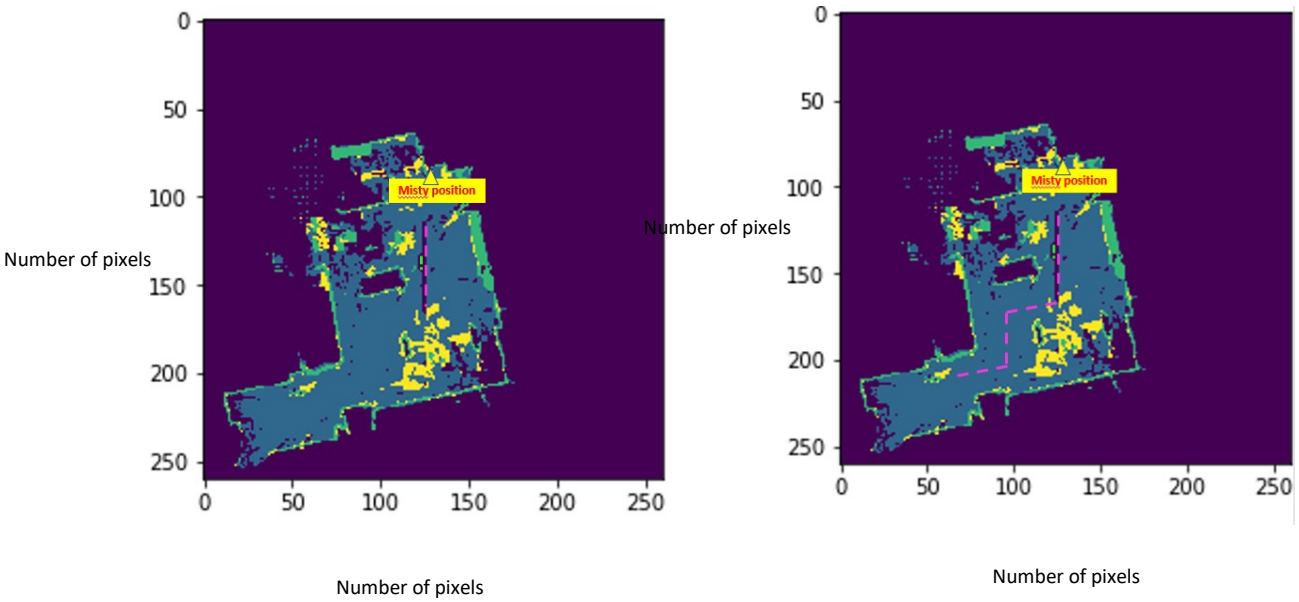


(A) (B)

Figure 40: panel A: the dushed purple line shows the linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm.

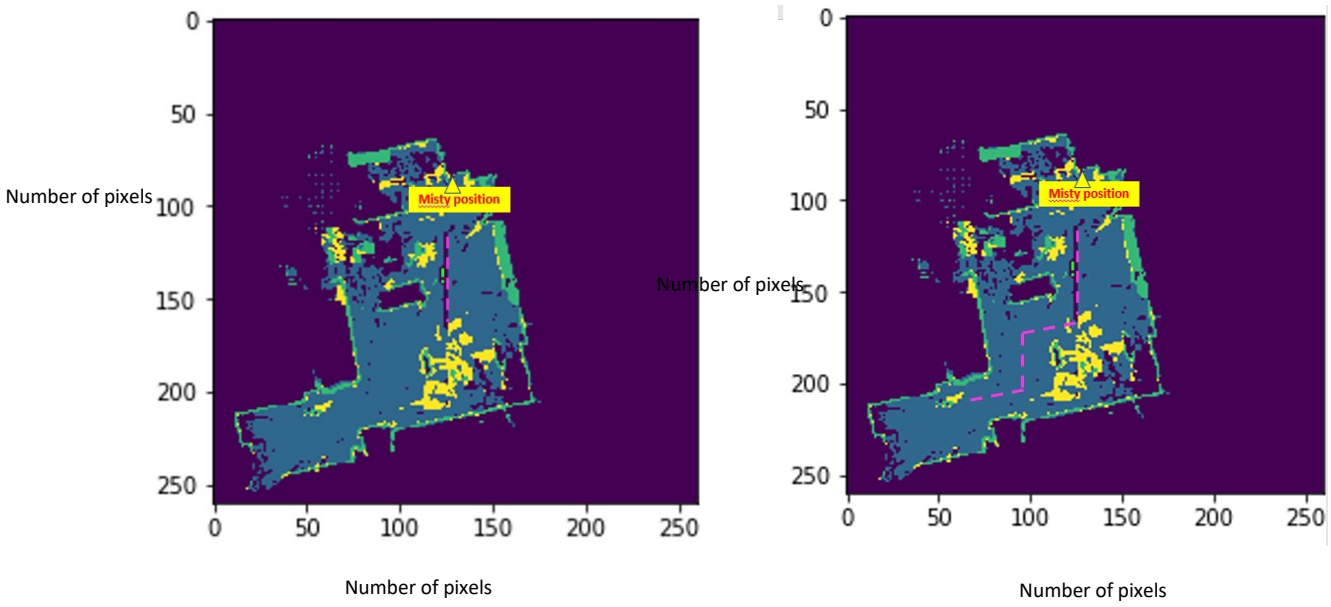
Panel B: the dushed purple line shows the non-linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm.

Figure 41 shows the tracking of the user performed when the robot is located on the table with a robot head of 20°, and the Target follows a path that is linear (panel A) and nonlinear (panel B). The dushed purple line represents the real path that the Target follows, instead the dushed green line represent the tracking performed by the algorithm. In Figure 42 is showed the tracking of the user performed when the robot head has an angle of 30° and when the Target follow a linear path (panel A) and a non- linear path (panel B). Also in this case the two dashed lines represent respectively the real path (purple) follow by the user and the path coming out from the algorithm (green). The Figure 43 shows the algorithm result when the robot head is at 40°. The colours of the legend have the same meaning of the two previous cases. Even if not in all cases the algorithm is capable to localize the user for the entire performed path, the combination of the described algorithm seems to be adequate in performing a localization and tracking of the subject.



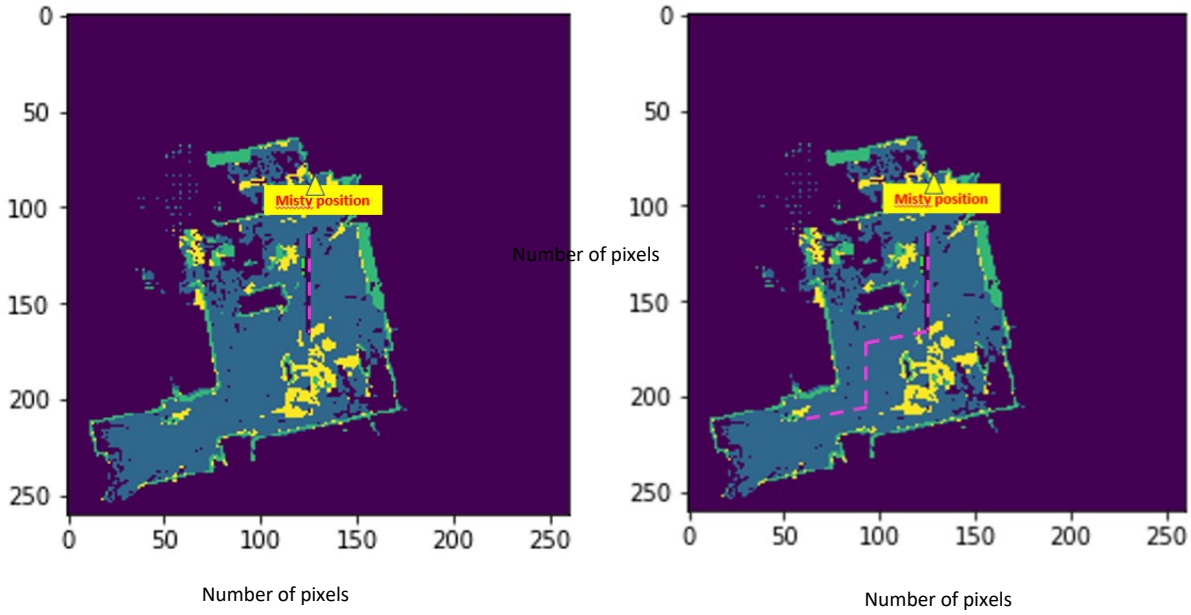
(A) (B)

Figure 41: panel A: the dushed purple line shows the linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm. Panel B: the dushed purple line shows the non-linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm



(A) (B)

Figure 42: panel A: the dushed purple line shows the linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm. Panel B: the dushed purple line shows the non-linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm



(A) (B)

Figure 43: panel A: the dushed purple line shows the linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm. Panel B: the dushed purple line shows the non-linear path performed by the user, while the dushed green line shows the map reconstructed by the Localization and Tracking algorithm

Chapter 4

Discussion

4. Discussion

The aim of this thesis is to implement an algorithm capable to localize and track a human in an indoor environment using on-board robot sensors, and to measure the algorithm's accuracy in performing this task. To accomplish this task, first a suitable robot with many sensors was selected. The robot chosen by the participants to the GUARDIAN project is the product of an American company, the Misty Robotics, which in 2019 realized a second version of a companion robot called Misty II.

Once the robot was chosen, in this thesis the robot sensors were explored and the measurement uncertainty of these sensors were analysed in order to select the most appropriate for user localization and tracking.

First, the depth sensor was studied. To extract the depth image from this sensor a reconstruction algorithm was necessary (section 2.3). The performances of this sensor were evaluated activating the sensor and critically analysing its results. As showed in Figure 2 and Figure 3, the reconstructed image obtained using this sensor had a very low quality. This condition was explained by the fact that the sensor used to capture the depth image is a laser. When the laser hit a dark surface, the absorption was high and so a lower signal return was provided. Moreover, when the laser hit a shiny surface the reflection of the signal was so high to bring the sensor to saturation. In addition, this sensor has a very limited working distance (0.5 m), which reduces the ability to track people in wider rooms; another critical issue concerns the activation of this sensor. In fact, no other sensors can be used when the depth sensor is in operation, limiting the capability of the whole equipment.

After depth sensor, the RGB sensor was studied. The performances of this sensor were evaluated using an existing algorithm for object detection, the YOLO algorithm. The tests performed with this algorithm helped to understand what Misty's RGB camera could see. The results suggested that the algorithm is able to detect person in a great variety of cases. Unfortunately use this algorithm for user detection and tracking would have meant that

in every moment the robot would have to acquire an image, causing some difficulties in user-acceptance due to some privacy concerns.

So, the research moved to a skill that is available on the robot, i.e. the Face Recognition skill. The algorithm, that runs when this skill is activated, returns as output a distance that expresses the distance between the sensor and the detected face. The decision to use this algorithm was based on the fact that even if a vision sensor is used, the Face recognition was not performed capturing images, so the privacy issue is minimized.

Once the method was worked out, an algorithm for the extraction of useful data from Face recognition skill was created. At this point, it was necessary to couple the output of face recognition skill with the environment mapping. Subsequently, the information on robot localization within the map is added. The robot-self localization information is provided thanks to the SLAM algorithm that runs on Misty.

After building an algorithm capable to extract these information from the robot, all the pieces were put together to build the localization and tracking algorithm.

An important step was to understand the accuracy in the distance measurement returned by the Face Recognition algorithm. For this purpose, some validation tests were performed.

Starting from the idea that the Localization and tracking algorithm could be integrated in the robot and that this latter must be a companion robot for the elderly, for the tests a population of different subject with different physical features was chosen. The subjects were put to some fixed distances in respect to the robot which was set in two possible scenarios. A first one in which the robot was located on the floor and a second one in which the robot was on table. An important parameter for these tests was given by the inclination of robot head. For this, three different angles of robot head are considered for each setup.

The data were collected repeating one time per subject the tests for each robot setup, robot head angle and distance robot-user. For each subject face recognition algorithm was activated for 36 times. The number of subjects selected was eleven and their height ranged between 164 cm and 194 cm.

After the acquisition tests the data were classified to perform a repeatability test and a global uncertainty test. For the first test a classification based the fixed distance was done, while for the second test the evaluation was based on the difference between input and output data (residues). From the first test came out that for distances ranged between 100

cm and 130 cm the best robot head angle is 20° for which the uncertainty is around 50 cm , while for distances ranged between 130 cm and 181 cm the best result was given for robot head angle equal to 30° which provides an average uncertainty of 40 cm. The same result was found for the two built setups. The only difference respect to the previous setup was the completely lack of results in certain subjects for same fixed distances, meaning that if the users were too tall and the robot head angle low, user could not be detected. The same condition occurred for lower user's height when the robot head angle was high. From the global uncertainty tests emerged instead that the best robot head configurations are 20° and 30° because for this value the lowest statistical confidence expressed in percentage was reached. In the first setup the statistical confidence for the 20° angle is of 51% while for the second setup a lower value (45.1%) is obtained for the robot head configuration of 30°.

Another proof was performed trying to find a relation between subject height and the standard deviation of the input values represented by the fixed distance at which users were located. This relation was studied for each robot head angle.

Against the expectancy from the results came out that no strong relationship exists among these two parameters because not a linear trend existed.

A further test was carried out by grouping the subjects by range of heights, but even in this case no relationship was found between subjects' heights and the standard deviation of the inputs.

After having performed all these tests and analysed the possible relationships between subject height, distance robot-subject and the angle of inclination of the robot head, the entire Localization and Tracking algorithm was tested. Before running the algorithm, a map of the environment was built. To construct the map, the robot must be moved by the user and the time required for this operation was of a couple of minutes. The raw data acquired during this procedure must be processed in Python virtual environment using an algorithm. To localize the robot in the map a tracking algorithm was necessary. The coordinates of the robot were added to Python reconstructed map. At this point the Face Recognition Algorithm could be activated.

As expected from the results on the tests performed on this algorithm, it was not able to track the user while he was performing a nonlinear path. This happens because Face Recognition algorithm requires that user be in front of the robot during the acquisition. Another limitation was given by the action range of this algorithm that was unable to

detect person at a distance more than two meters. Because the nonlinear path was at a distance from the robot higher than this value, in the result there was a missing of the tracking in that regions.

The worst results are obtained when the robot was located on the floor both in the case in which the angle of the robot head was set as small (20°) and in the case in which the angle of the robot head was set as high (40°) . This suggests that the angle of the robot head cannot be a fixed parameter during data acquisition and therefore during the entire tracking of the person in the room.

From linear path results with the robot located on the table come out that the algorithm was able to track better the user. This occurred because the sensor was positioned at a height closer to that of the target unlike what happened when the robot is placed on the floor.

Even if these results were better than those obtained for robot located on the floor, I encountered in a problem regards the obtaining of robot pose. In fact, when the robot was located on the table the tracking algorithm was not capable to obtain robot pose. This problem in this thesis was overcame considering the position occupied by the robot on the table with the same coordinates of the robot located on the floor. This solution was acceptable because the map was reconstructed on the x-y plane while the elevation occurs in the z plane.

Chapter 5

Conclusion and Future work

5. Conclusion and Future Work

A fusion technique between an infrared sensor and a vision sensor system was applied on a robot to obtain an internal tracking system for use in living environments.

The infrared system is used to build a map of the environment in which the robot is moving and to obtain its exact position and orientation on the map.

The vision system is used by a facial recognition algorithm to extract distance information from the robot to the user's face. By combining these sensors, a location and tracking algorithm was created.

Tests were carried out on the facial recognition algorithm to measure if the distance data obtained from it are accurate.

The results showed an average measurement uncertainty of 50 cm when the robot is placed on the table and when its head angle is between 20° and 30°, regardless of the height of the users. However, when the robot is on the ground, a different scenario appears, as the algorithm is not always able to trace the face of a subject. A problem that emerged from this study concerns obtaining the position and orientation of the robot when it is placed on the table. In fact, in this condition the robot is unable to obtain the information relating to its placement in the map since this latter was acquired when the robot is on the floor.

In conclusion, from this thesis it emerged that the fusion of a visual sensor with an infrared sensor on the Misty robot represents an encouraging starting point for the construction of a truly efficient person localization and tracking algorithm.

A next step in achieving satisfactory results could be the fusion of these two sensors with sound sensors. This latter in fact could be used to move the robot in the direction from which typical domestic noises come, increasing the possibility of capturing the human face with the face recognition algorithm even when the subject is not positioned in front of the robot.

References

[1] Irene Castro, “Indoor human localization: a sensor fusion approach using long distance capacitive and infrared sensors”, Rel. Mihai Teodor Lazarescu, Luciano Lavagno. Politecnico di Torino, Corso di laurea magistrale in Ingegneria Elettronica (Electronic Engineering), 2020

[2] https://www.bbc.co.uk

[3] Dal Ben, Sebastiano & Fontanelli, Daniele. “Majority Effect in Cooperative localisation of Mobile Agents using Ranging Measurements”, 1-6. 10.1109/, 2020.

[4] Zhihua Wang, Zhacochu Yang and Tao Dong, “A Review of Wearable Technologies for Elderly Care that Can Accurately Track Indoor Position, Recognize Physical Activities and Monitor Vital Signs in Real Time”, Sensors, vol.17, 2017.

[5] Shamsfakhr, Farhad & Palopoli, Luigi & Fontanelli, Daniele & Motroni, Andrea & Buffi, Alice, “Robot Localisation using UHF-RFID Tags for Industrial IoT Applications”, 659-664. 10.1109, 2020.

[6] Álvarez-Aparicio, Claudia, Ángel Manuel Guerrero Higueras, F. R. Lera, Jonatan Gines Clavero, Francisco Martín and Vicente Matellán Olivera. “People Detection and Tracking Using LIDAR Sensors.” Robotics 8 (2019): 75.

[7] P. Zhao et al., "mID: Tracking and Identifying People with Millimeter Wave Radar," 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini Island, Greece, 33-40, 2019.

[8] Xueling Luo et al, “A Real-time Moving Target Following Mobile Robot System with Depth Camera” IOP Conf. Ser.: Mater. Sci. Eng. 491, 2019.

[9] Sales, Francisco & Portugal, David & Rocha, Rui., “Real-time People Detection and Mapping System for a Mobile Robot using a RGB-D Sensor” ICINCO Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics,2014.

[10] Truong, Quang & Ngo, Ha Quang Thinh & thanh phuong, Nguyen & Nguyen, Hung., “Control of Mobile Robot to Track Target by Using Image Processing” 1-5. 10.1109, 2019.

[11] Antonucci, A., Magnago, V., Palopoli, L., & Fontanelli, D., “Performance Assessment of a People Tracker for Social Robots”, IEEE International Instrumentation and Measurement Technology Conference (I2MTC), 1-6, 2019.

[12] Medina, C.; Segura, J.C.; De la Torre, Á., “Ultrasound Indoor Positioning

System Based on a Low-Power Wireless Sensor Network Providing Sub-Centimeter Accuracy", Sensors, 13, 3501-3526, 2013.

[13] Qu D., Bo Yang, Nanhao Gu, "Indoor multiple human targets localization and tracking using thermopile Sensor", Infrared Physics & Technology,Volume 97, pp.349-359, March 2019.

[14] Ha Manh Do et al. "RiSH: a robot integrated smart home for eldely care", Robotics and Autonomous Systems, pp.74-92, 2018.

[15] Jiang C., Fahad M., Guo Y., Chen Y., "Robot-Assisted Smartphone Localization for Human Indoor Tracking", Robotics and Autonomous Systems, pp.2-15, 2018

[16] Yang S., Hans A., Zhao W., Luo X., "Indoor Localization and Human Activity Tracking with Multiple Kinect Sensors", Computer Communications and Networks. Springer, Cham,2020.

[17] Mengmeng W. et al., "Real-time 3D Human Tracking for Mobile Robots with Multisensors", EEE/ASME Transactions On Mechatronics", vol. 23, no. 3, June 2018.

[18] Halima I, Laferté JM, Cormier G, Fougères AJ, Dillenseger," Depth and thermal information fusion for head tracking using particle filter in a fall detection context", 2020.

[19] Halima I, Laferté JM, Cormier G, Fougères AJ, Dillenseger 625 JL. "Sensors fusion for head tracking using Particle filter in 626 a context of falls detection", First International conference 627 on signal processing & artificial intelligence (ASPAI' 2019); p. 134-139., 2019.

[20] Singh, Roshan & Khurana, Rajat & Kushwaha, Alok & Srivastava, Rajeev, "Combining CNN streams of dynamic image and depth data for action recognition" Multimedia Systems, 2020.

[21] A. Rudenko, T. P. Kucner, C. S. Swaminathan, R. T. Chadalavada, K. O. Arras and A. J. Lilienthal, "THÖR: Human-Robot Navigation Data Collection and Accurate Motion Trajectories Dataset," in IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 676-682, April 2020.

[22] Shenoi, Abhijeet & Patel, Mihir & Gwak, JunYoung & Goebel, Patrick & Sadeghian, Amir & Rezatofighi, Hamid & Martin-Martin, Roberto & Savarese, Silvio, "JRMOT: A Real-Time 3D Multi-Object Tracker and New Large- scale Dataset", March 2020.

[23] Sankar, S.; Tsai, C.-Y., "ROS-Based Human Detection and Tracking from a Wireless Controlled Mobile Robot Using Kinect", Appl. Syst. Innov. 2019.

[24] Privandoko,Gihih; Wei,choi kah; Achmad, Muhannad Sobirin Hendrivawan, “Human following on ROS framework: a mobile robot”. Sinergi s1, v.22, n.2, p. 77-82, June 2018.

[25] Algabri, Redhwan & Choi, Mun-Taek., “Deep-Learning-Based Indoor Human Following of Mobile Robot Using Color Feature”, Sensors. 20., 2020.

[26] Yan, Z., Duckett T., Bellotto N., “Online learning for 3D LiDAR-based human detection: experimental analysis of point cloud clustering and classification methods”, Auton. Robot 44, pp. 147–164, 2020

[27] Kwon, S.; Park, T., “Channel-Based Network for Fast Object Detection of 3D LiDAR”, Electronics 2020.

[28] https://www.mistyrobotics.com

[29] Angulo Bahón, Cecilio; Velasco García, Manel, Master thesis, 2016-09-05.

[30] Robotics Penn State