Universidad de Granada

Escuela Técnica Superior de
Ingenierías Informática y de Telecomunicación
Departamento de
Ciencias de la Computación e Inteligencia Artificial
Programa de Doctorado:
Diseño, Análisis y Aplicaciones de
Sistemas Inteligentes

DOCTOR OF PHILOSOPHY DISSERTATION

A Stereo Vision System based on Soft Computing Techniques for Human Robot Interaction

by

Rui Paúl Oliveira

Editor: Editorial de la Universidad de Granada
Autor: Rui Paúl Oliveira
D.L.: GR 526-2014
ISBN: 978-84-9028-833-7

THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

This Thesis was submitted to the University of Granada
in accordance with the criteria necessary for the award of the Doctorate Degree

Organization:

Departamento de Ciencias de la Computación e Inteligencia Artificial; Granada; España

Title:

A Stereo Vision System based on Soft Computing Techniques for Human Robot Interaction *(Un Sistema de Visión Estéreo Basado en Técnicas de Soft Computing para la Interacción entre Humanos y Robots)*

Author:

Rui Paúl Oliveira

Supervisors:

Dr. Eugenio Aguirre Molina (Universidad de Granada, España)
Dr. Miguel García Silvente (Universidad de Granada, España)
Dr. Rafael Muñoz Salinas (Universidad de Córdoba, España)

El doctorando D. Rui Filipe Paúl Miranda de Oliveira y los directores de la tesis Dr. Eugenio Aguirre Molina, Dr. Miguel García Silvente y Dr. Rafael Muñoz Salinas garantizamos, al firmar esta tesis doctoral, que el trabajo ha sido realizado por el doctorando bajo la dirección de los directores de la tesis y hasta donde nuestro conocimiento alcanza, en la realización del trabajo, se han respetado los derechos de otros autores a ser citados, cuando se han utilizado sus resultados o publicaciones.

Granada, 27 de Junio del 2013.

Directores de la Tesis

Doctorando

Fdo.: Eugenio Aguirre Molina

Fdo.: Rui F. Paúl Miranda de Oliveira

Fdo.: Miguel García Silvente

Fdo.: Rafael Muñoz Salinas

To my parents, sister and to Rachel

Contents

Acknowledgements

The first words of this dissertation are dedicated to all people that contributed, in a way or another, to the accomplishment of such an important milestone in my life. Their "human to human interaction" are for me an inspiration regarding what "human-robot interaction" should become.

To start, I would like to thank my advisors Eugenio Aguirre Molina, Miguel García Silvente and Rafael Muñoz Salinas for giving me the privilege of working with them and for all precious advices towards the achievement of my goal. Their support and encouragement proved to be crucial especially during the hardest moments of this path.

My sincere gratitude to all my friends and family that, even without noticing, contributed to this goal. It was, many times, their words, actions and "jokes" that made me face my worries with a much more positive attitude.

A special word also to all researchers that contribute with their daily work and efforts to make science advance. They inspire me with their ideas and it is motivating to know that many spend hundreds of hours per year trying to make a better world.

Finally, I cannot express in words how grateful I am to Rachel, my mum, dad and sister. I owe most of what I am to them so this work is also a part of them. A big kiss to them.

This work has been partially funded by the FCT (Fundacao para a Ciencia e a Tecnologia) Scolarship SFRH/BD/22359/2005, POPH/FSE (Programa Operacional Potencial Humano and Fundo Social Europeu) as well as the Andalusian Regional Government project P09-TIC-04813.

Abstract

The main goal of this thesis is the development of visual techniques that could be useful in order to establish a natural interaction between people and robots. In this context, “natural” interaction means an interaction similar to the ones existing between humans. Therefore, efforts were put in making it possible for a robot equipped with a Stereo Vision (SV) system to study and analyse the behavior of those people which are located in its surroundings.

The motivation behind this goal is to give robots the ability to behave and choose between actions as any human would do. This means performing several tasks such as: being able to detect and track people on the surroundings of the robot and accurately detecting who is potentially interested on the actions executed by the robot and/or responding to them. Furthermore, by doing so, robots may use their resources more adequately and even improve their decision capabilities and communication methodologies while achieving a kind of behavior similar to the human behavior.

To achieve this kind of Human-Robot Interaction (HRI) different techniques are detailed. These techniques contribute to solve several issues inherent to this field. In particular, Soft Computing (SC) techniques are employed to deal with uncertainty and vagueness as well as to represent variables and rules in a human oriented way. Image analysis techniques are also employed to extract relevant information from the scene. All of them allow the enhancing of the socialisation of robots.

The purpose of this work is twofold. First, detection and tracking of people that are located in the surroundings of the robot, are done. Second, computing whether a person is interested in interacting with the robot, requesting its attention or responding to its actions, is carried out. This is done by analysing typical interaction cues between humans such as: the distance between interlocutors, head pose, arms shaking, head shaking/nod-

ding and smiling.

To achieve the first goal, two different methods are presented: one based in a probabilistic approach and a second one based on a “possibilistic” approach. The probabilistic method presents a novel approach for person tracking which combines depth, color and gradient information based on stereo vision. The degree of confidence assigned to depth information in the tracking process varies according to the amount of stereo information found in the disparity map. A novel confidence measure is defined for it and the tracking is carried out using Particle Filter (PF) techniques. The second method, based on a possibilistic approach, is employed to add more information based on expert knowledge, when evaluating the particles, and without being confined to the probabilistic models. This approach also uses Fuzzy Logic (FL) when managing stereo information in order to improve the people detection phase. Thus, in the people detection phase, two fuzzy systems are used to filter out false positives of a face detector. Then, in the tracking phase, a new Fuzzy Logic based Particle Filter (FLPF) is proposed to fuse stereo and color information assigning different confidence levels to each of these information sources. Information regarding depth and occlusion is used to create these confidence levels. This way, the system is able to keep track of people, in the reference camera image, even when either stereo information or color information is confusing or not reliable.

Considering a robot as an intelligent system, the determination of some typical interaction situations is an interesting ability to implement. Therefore, to achieve the second goal, a method based in several cues, namely the distance and angle towards the robot and the person head pose, is presented. The head pose is estimated in realtime by a view based approach using Support Vector Machines (SVM) while a Fuzzy System(s) (FS) is used to compute the final interest value, based on the three mentioned variables. Whenever the level of interest achieves a high value, the person is analysed in more detail to detect the position and the motions of the arms as well as whether the person is shaking or nodding the head. This information is managed by a fuzzy system in order to detect a possible interest demand or the intention of the person to say yes or no using his/her head. Some of the above mentioned sources of information are used together with smile detection, in the last work mentioned in this thesis, to build a system based on FL which is able to measure certain types of human response. As the reliability of the visual information detected by the system mainly depends on the distance of the person towards the camera, we prioritise different visual cues according to the distance of the user towards the robot. The human

response is computed by means of a hierarchical fuzzy system that is able to deal with the uncertainty and vagueness of the measures depending on the distance of person. This human response measure is used for detecting the person or people which are responding to the social interactions proposed by the robot and it might be also used to improve or adjust the interaction skills of the robot in the future.

Keywords:

People Detection, People Tracking, Interest Detection, Human Robot Interaction, Human Response, Fuzzy Logic, Stereo Vision, Particle Filter

List of Abbreviations

AI	Artificial Intelligence	1
ANN	Artificial Neuronal Networks	38
CPU	Central Processing Unit	31
EC	Evolutionary Computing	10
FLPF	Fuzzy Logic based Particle Filter	78
FL	Fuzzy Logic	135
FS	Fuzzy System(s)	136
GA	Genetic Algorithm(s)	3
GPS	Global Positioning System	8
HCI	Human-Computer Interaction	70
HFS	Hierarchical Fuzzy System(s)	136
HRI	Human-Robot Interaction	133
ML	Machine Learning	173
MPF	Multi Particle Filter	26
PCA	Principal Component Analysis	173
PF	Particle Filter	134
ROI	Region of Interest	35
SC	Soft Computing	133
SV	Stereo Vision	136
SVM	Support Vector Machines	136

List of Figures

List of Tables

Preamble

For an Artificial System to act in a "natural" way, there are certain steps that should be performed analogously to the way humans behave. In this work, it is assumed that the robot is completely immobile while the process of interaction is carried out. The same condition applies to the stereo camera which is static during that process. This supposition may be considered acceptable as typically, a person who is analysing the behaviour of other people and trying to understand their reaction to his or her interactions, as it is the case of our robot, is usually immobile and paying attention to his or her interlocutors. Obviously, that person may start moving during an interaction, but it is not the aim of this work to make it possible for the employed robot to move and to analyse the possible interaction demands at the same time. Nevertheless, although the robot is immobile during the interaction process, people may freely move in its surroundings.

The first step which is taken into consideration is the ability to correctly detect and track people in its surroundings. People detection and tracking can be done in various ways and with different kinds of hardware. When computer vision is used, the system analyses the image and searches for cues that provide important information for the detection of people. Those cues could be, for instance, morphological characteristics of the human body ([HM03]) or dynamic skin colour models([SSA04]).

Nowadays, several methods employed for tracking people are based on the colour information available from people cloths. Commonly, the first step is to create a colour model of the person to be tracked. Then, throughout a sequence of images, the position and size of the image region whose colour model best matches the person colour model, is considered the new position and size of that person. This technique is called adaptive tracking and it is especially appropriate for tracking non-rigid targets, of which there is no explicit model, or when the background estimation is not possible.

As most of these techniques rely uniquely on colour information, they present several drawbacks. The most important is the confusion between two or more image areas that have the same colour distribution when they are close to each other. Because there is no other information to distinguish them, this issue can cause the system to confuse the objects being tracked. This confusion can also happen with the background, if the tracker does not have information about which parts in the image are part of the background. In case background, or a part of it, has a colour distribution similar to the person being tracked, the target can be lost. Finally, the situation where the tracker assigns a subregion of the tracked person as the whole region of the tracked person, may also happen. That becomes a problem when determining the appropriate size of that person in the camera image, as a part of the body of the person is identified as the whole person.

Some authors have proposed the use of stereo technology which nowadays has been thoroughly studied and has become more common in computer applications ([BBH03]). With the development of well consolidated technologies and commercial hardware that deal with stereo computation issues, this technique has turned out to be an important tool when developing computer vision applications such as tracking algorithms. These algorithms can take advantage of pixel distance information for solving problems that non-stereo tracking algorithms present. Firstly, the possibility of knowing the distance from the camera to the person can be of great help when tracking is taking place. Secondly, distance information is less sensitive to illumination changes than information provided by a single camera.

As soon as the problem of people detection and tracking is solved, other problems should be studied in order to develop a robot with social capabilities. By reading [FND03], one can acquire a general idea of how different approaches contribute to solve the complex problem of making up a social robot. As described in this reference, the range of issues that one has to take into consideration varies from the design of the robot itself to its acceptation by the society, from the detection of emotions to the expression of the robot emotions, from the possibility of simulating a personality to the imitation of other personalities. In all cases, a social robot should be prepared to communicate and analyse communication cues from its interlocutors to better accomplish these tasks. And it should do it the most natural way, using natural cues.

Although different authors have contributed with several papers on this field, as cited in [FND03] and in Section 1.3, there are still a wide range of cues to explore which allow a more natural interaction between social robots and humans. In this work, efforts are centred in recognising when and how long a person is interested in establishing an interaction as well as in determining the level of response of those people to the social interactions proposed by the robot. In this task, several types of signals from the human can be taken into account (both verbal and non-verbal). Some authors [BFJ+05a] use sound source localization or speech recognition combined with visual perception to detect which people are the most interested. In other cases, facial expressions [SKKB01] or hand gestures [GNS+02] are analysed. In this work a special interest is put in the analysis of a set of typical interaction situations that can be integrated in a more complex system in the future.

This work is based in one of the most important human senses used by humans to percept its surroundings and to recognise interaction cues: vision. In addition, in order to properly work, this sensor does not oblige the robot interlocutors to wear any special sensor, making it similar to the human way of perceiving actions. These different visual features are handled by using another similar to human way of reasoning: fuzzy logic. More specifically, various cues were defined in this work, which could naturally indicate whether one or several people are interested or collaborating and responding to an interaction. As visual cues to infer about interest detection and human response, we use each detected person head pose towards the robotic system, their arms shaking movement, as well as their smile. Furthermore, in our opinion, people who try to interact with another person avoid being occluded by other people or objects. In addition, a technique to infer about basic shaking and nodding of the head (which might give visual clues about an answer to a “YES” or “NO” type of question) is also presented. As the reader may notice, these are “natural” types of cues that are used on the daily routine of every person life when interacting with each other.

In order to achieve our goals, several kind of restrictions had to be taken into account. For instance, it is almost impossible to detect a smile when someone is located at more than a couple of meters from the camera. Another restriction is related to the impossibility to capture arm movement whenever someone is too close to a fixed camera, as a considerable part of the body is located outside the field of view of the camera. Therefore, it is simply to understand that distance plays a key role when choosing which visual features to privilege at each time. Thus, in the system hereby described, the human response is computed by means of a hierarchical fuzzy system that is able to deal with the uncertainty and vagueness of the measures depending on the distance of person. By measuring this response, the robot is potentially able to interact more naturally and to improve the proposed activity.

Major Contributions

The work described in this thesis contributed with several advancements in the area of Human-Robot Interaction (HRI). Different methodologies were tested which may continue to be improved in the upcoming years. The main contributions of this thesis are:

1. A novel method for people tracking based on Particle Filter (PF) that integrates depth, color and gradient information to perform a robust tracking. Since depth information cannot be always extracted because of occlusions or absence of texture, our method is able to deal with this problem by defining a certainty measure that indicates the degree of confidence in depth information.

   This contribution has been published in [MSAGSP07].

2. A system capable of detecting and tracking various people using a new approach based on color, Stereo Vision (SV) and Fuzzy Logic (FL). Initially, in the people detection phase, two fuzzy systems are used to filter out false positives of a face detector. Then, in the tracking phase, a new Fuzzy Logic based Particle Filter (FLPF) is proposed to fuse stereo and color information assigning different confidence levels to each of these information sources. Information regarding depth and occlusion is used to create these confidence levels. This way, the system is able to keep track of people, in the reference camera image, even when either stereo information or color information are confusing or not reliable.

   This contribution has been published in [PAGSMS12].

3. A new fuzzy system that allows the visual detection of possible interaction demands and the shaking or nodding of the head. The level of interest of a person to interact with the robot is calculated by analysing his/her position, the pose of his/her head and their arms shaking movement. The head pose is estimated in realtime by a view based approach using Support Vector Machines (SVM).

   This contribution has been published in [AGSG$^+$07].

4. A system capable of measuring human response from people located in the surroundings of a social robot using FL and SV. To achieve this goal, the system analyses different visual cues which humans "naturally" use on their daily routines and which may supply a feedback to the activity proposed by the robot. The human response is computed by means of a Hierarchical Fuzzy System(s) (HFS) that is able to deal with the uncertainty and vagueness of the measures depending on the distance of person.

   This contribution has been submitted for publication in the International Journal of Human-Computer Studies (IJHCS).

Thesis Structure

This thesis is organised is several chapters, starting by an Introduction in Chapter 1. Chapter 1 presents a review of the state of the art on different fields of HRI in its first two sections. In its third and last Section, several papers which address various topics related to this work are commented. In Chapter 2, a description of the system configuration used on the different parts of this work is given, together with the basis of some of the used techniques, namely stereo and colour modelling, Principal Component Analysis (PCA), SVM and FL.

From Chapter 3 to 5 this work most important contributions are presented. On Chapter 3 two detection and tracking methods are presented. The first one is primarily based on a probabilistic approach while the second one is based on a "possibilistic" approach. On Chapter 4 a method for detecting interest and attention request is presented. Finally, on Chapter 5, the approach for human response detection is presented.

The concluding Chapter 6 features the conclusions and final considerations while suggesting possible works to improve or continue the current work.

Chapter 1

Introduction

Robotics and Artificial Intelligence (AI) are two different areas that, day by day, are getting more connected to each other. Nowadays, it is possible to see that robots are being assigned several kinds of tasks, in different areas of our society. Most of these tasks could only be performed by humans, because they required a certain know-how and intelligence that machines didn't have. However, the evolution of hardware, the results obtained in AI research and its application to robotics, allowed robots to perform these kinds of tasks. In the scientific community, researchers are attracted by the fact that a machine can not only move in a real environment but also behave, precept and think like a human does. In [Bro91] and [Yan08] it is possible to read a review on different research works in the area of robotics. It is also possible to read some comments about their influence in the area of AI as well as to observe the relationship between last year's focus of research in AI and the subsequent developments in robotics. To sum up all these ideas, we can say that "Robotics is where Artificial Intelligence meets the real world, and supplies the necessary experience to validate any system".

A robot should be able firstly, to detect and understand the environment around it, and secondly, to act according to such perceptions and the goals that it was conceived to. As a first challenge, it is necessary to make robots precept their environment (which may be continuously changing), using the different kinds of sensors at our disposal. However, sensors are affected by errors and uncertainty. The second challenge is to use sensor information in order to make robots able to act in their environment. At the same time, they should accomplish their tasks in the most effectiveness way by choosing, among several previously unknown options and possibilities, the most adequate one.

To achieve such a complex task, the designer of the control system of the robot should develop a system able to continuously extract information from sensors and to supply the necessary actions to the actuators, in real time. Such a system is probably made of several different subsystems (like artificial vision systems, trajectory generators, hardware controllers for sensors and actuators, etc) working and exchanging information between themselves. Thus, the designer should also be concerned about their integration as a whole system, which implies tasks of coordination between the different parts and the possibility of adding or changing the existent ones.

The definition of such robotic systems becomes more complicated as the tasks assigned to the robot become more complex. This is due to the amount of different agents and subsystems that are part of the whole system and are essential for the robot to achieve its goal. Another problem is the diversity of hardware pieces, sensors, actuators, control drivers, communication protocols, among others, that are also part of the system. Furthermore, the difficulty of building robot systems increases when real time and robustness constraints are required.

Research in this area has been mainly focused in autonomous indoor and outdoor navigation, new software architectures for autonomous robots, planning, manipulation and grasping, learning, perception, Human-Robot Interaction (HRI) and robot-robot interaction. The results of research in these fields have allowed the development of robots for several different applications like building cleaning, object grabbing, security and surveillance, inspection, agriculture, garbage treatment and collection, submarine exploration, planetary exploitation, among other areas.

Several authors have tried to put into a few words the definition of an autonomous robot. An interesting definition is the one given by Ronald C. Arkin [Ark98]: “An intelligent robot is a machine capable of extracting information from its environment and, using its knowledge about the world, act in a coherent and intended way”. Alessandro Saffiotti also resumes in a few words the goal of mobile robotics in [Saf97]: “The goal for an autonomous mobile robot is to be able to move and to achieve its goal without human aid in real world environments that were not specially designed for them”.

In the next paragraphs, both the areas of robotics and HRI will be introduced more in detail. Some fundamental topics in robotics like environment representation, sensors and actuators, control architectures as well as the contribution of Fuzzy Logic (FL) in this field, are first described. Then, the use of computer vision applied to robotics is also emphasised as this is our system environment main perception method. Secondly, we will focus in HRI as well as known Soft Computing (SC) methods employed in this area.

1.1 Autonomous Robots

In order to make robots behave like humans in a real environment, it is important that they can understand the information around them, process it, and act the most adequate way. To achieve this complex task, different systems take part in the process. Environment perception has to be made by means of sensors. These sensors could be considered the equivalent to the basic five senses of the human being: "sight, hearing, smell, taste and touch". As said before, sight or vision is the main sense used in our system, performing an important role in the robot perception of the real world. Therefore a more detailed description is given in the last part of this section. Then the robot has to represent the sensed environment and decide what to do according to the environment and its objectives. These decisions are made by the control system that indicates to the actuators what to do at each moment. These systems can learn and sometimes be integrated by means of SC methods like Genetic Algorithm(s) (GA), FL, etc.

1.1.1 Environment Perception

As previously indicated, robots have a sensing system that allows them to extract important information from the environment. Sensors often produce errors due to noise or to their own limitations. In [JF93] it is possible to find some information about the characteristics and the properties of sensors employed in mobile robotics. Each sensor has limitations that influence the maximum distance at which they can work at, depending on their nature. A brief explanation about the different kind of sensors available is given in the next paragraphs:

- Ultrasonic sensors allow the detection of the echo of a sonar signal with a precision which is enough for detecting obstacles (1% error) and a range varying from 20 centimetres to 6 meters. It has some drawbacks like the lack of accuracy when estimating the direction of the detected object and the non reflection or non perpendicular reflections by the objects in the environment. Besides, they present problems of echo detection when objects are too close from the sensor.

- Infrared sensors use an infrared light source instead of an ultrasonic source and are capable of a higher angular precision regarding ultrasonic sensors. They are also capable of detecting closer objects. However their behavior is unstable in the presence of other light sources like direct sun light. They also have a shorter range while needing a previously and precise calibration phase.

- Laser sensors work like infrared sensors but with a laser signal. They allow a higher angle precision and they are able to detect objects within a larger distance.

- Touching sensors are able to detect collisions, although it could be considered as a sensor of non desirable states because one does not want the robot to be colliding with walls.

- Gyroscopes, accelerometers and compasses allow orientation and acceleration detection, although sometimes they lack in accuracy.

- Visual information is one of the most important sensors that a robot can have. Maybe it is even the most important for the human being when interacting with other human beings, animals or objects. A lot of work has already been done to improve this kind of sensors. At the end of this section it will be given a deeper description about the vision sensor.

- Microphones are able to detect sound sources although it is still very difficult to detect with precision the position of those sources. Nevertheless, it is already possible to distinguish the sounds and even to detect what one is saying by using speech recognition software.

  More information about the characteristics of the different available sensors can be found in [JF93] and [SN04]. In [Spe13], the novelties and updates concerning robot sensors and actuators can be found.

1.1.2 Computer Vision

Computer vision allows computers to understand the meaning of the multidimensional data existing in images. Image data can have color or grayscale information, and can be analysed as a stereo image (two or more images of the same scene), video sequence, 3D images, etc.

Computer vision is usually studied as an area of AI, where usually one image is supplied, instead of text, as the input for a given system, with the purpose of controlling the behavior of that system. Some of the learning methods used in computer vision are based in learning techniques developed under AI. A vision system can be divided into six different systems:

- Knowledge base: representing the knowledge about the problem to solve using vision. It could be something like detailing the interest regions of some image, reducing the area where a specific search is going to be done.

- Image acquisition: system responsible for acquiring the image. It can be a camera, radar, sonar, etc. As Stereo Vision (SV) is the main acquisition method of the works presented in this thesis, more detail will be given in this section and in Section 2.2.1 about this type of sensor.

- Image processing: system that processes the image for some objective. Examples are filtering, resampling, decomposition in frequencies, border detection, estimate disparity in stereo images, etc.

- Segmentation: consists in separating the objects or parts of the image. It is usually based in discontinuity or similarity criteria.

- Representation and description: usually comes after segmentation and consists in representing one region using its internal (interior) or external (border) characteristics and describing it according to some representation method previously adopted. An example of internal representation is Principal Component Analysis (PCA).

- Recognition: is the process of classifying and labelling some object. This classification can be done using Artificial Neuronal Networks (ANN), Support Vector Machines (SVM), statistical methods, etc.

For more information about these systems, readers are referred to [RC08].

Stereo Vision

Stereopsis is the process in visual perception that allows the measurement of the depth or distance of the objects in the image. Depth from stereopsis is possible because of the slightly different positions that each of the human eyes occupies in a human head. It was discovered by Charles Wheatstone in 1833, when he found out that each one of a person eyes sees the world from slightly different places. This way, both projections of each visualised object are placed slightly differently in the horizontal axe, providing information about the depth of the object. Wheatstone proved that the distance in the horizontal axe, called disparity, was responsible for the feeling of depth.

This idea is the same used in SV. Two cameras capture slightly displaced images that after a matching process are able to supply the information about depth of some of the points in the scene. Readers that are interested in obtaining more information about this process are referred to [BBH03] and [Tor11]. Section 2.2.1 is also dedicated to SV.

Vision and Robotics

Vision is probably the most important sensor for humans and robots. It is able to retrieve very useful information that can be of primordial importance for robots. For that reason, vision has been applied to different problems in robotics, like navigation, people and object detection, etc.

In particular, research in computer vision applied to robot navigation has been carried out. Navigation systems that use vision can be classified into two categories: Visual guided navigation in interiors or in exteriors. A survey about this field can be found at [DK02] and a recent book devoted to the theory and development of autonomous navigation of mobile robots using computer vision based sensing mechanism is available at [AC13].

Vision is also of great importance for detecting and tracking people in the surroundings of the robot and allows it to extract different features that can be used to detect the level of attention, gestures, specific movements, etc. The extraction of these features is of great importance for the establishment of a natural interaction between robots and humans and for choosing the adequate behavior towards those people. In section 1.1.7, a brief review of the use of vision in HRI is given.

1.1.3 Environment Representation

To allow robots to move autonomously in their environment and to know their position, it is desirable that they have some way of representing the environment. These maps could be designed using three main approaches:

- Geometrical approaches [Elf89] are based in 3D models or cell maps for environment representation. Cell maps are made of cells with a previously assigned size and an occupancy value that tell whether there is an obstacle or not.

- Topological approaches [KW94] are based upon the detection of several features in the environment and their relationships. Graphs are used to represent this relationships and this kind of approach is much more abstract than the geometrical one.

- Hybrid approaches [AG02] combine the advantages of both geometrical and topological approaches.

1.1.4 Actuators

Actuators are mechanisms that allow robots to react accordingly to the different events occurring in their surrounding environment. A robot may have different kinds of actuators, responsible for different tasks. Most of them are used for locomotion and manipulation tasks. The most common are made from electrical motors capable of generating movement from electricity. In the book of Jones and Flynn [JF93], it is possible to find information about the most used locomotion systems used in robotics. There are other kind of actuators like hydraulic levers, pneumatic actuators, hydraulic pistons, relays, comb drive, piezoelectric actuators, thermal bimorphs, etc., according to the type of application to be executed. As previously mentioned, in [Spe13], the novelties and updates concerning robot sensors and actuators can be found.

1.1.5 Localization

One of the main problems in the development of mobile robots is the knowledge about the position of the system. Mobile robots should be able to know their position so they can achieve their tasks, namely navigating in the environment. Precision required in localization depends on the application the robot is performing and on the control architecture employed. While some applications require exact location of the vehicle, others work with approximated values. For instance, while geometrical environment representation approaches require precise location of the vehicle, topological approaches usually do not need such precision. Many authors have proposed different techniques to solve this problem. A brief explanation about some of them will be given in the next paragraphs. Readers interested in obtaining more information about these techniques are referred to [Wil97].

- Odometry

Odometry is the most used localization system used in robotics for its low cost and its fast computing time. It is based in the measurement of the linear distance traversed by the wheels of the robot. The main drawback of this approach is the errors produced by the slipping of the wheels.

- Inertial navigation

  Inertial navigation uses gyroscopes and accelerometers to measure the rotation and the acceleration of the robot during its movement. Accelerometers are still very sensible to inclination making them produce some errors. Gyroscopes are nowadays more precise and cheaper than some years ago and allow the correction of errors that appear in odometry.

- Magnetic Compass

  Magnetic Compass gives the robot the possibility of orientation according to the earth magnetic field. The main disadvantage of this approach is the distortion due to electrical wires existing everywhere. This makes the use of this orientation method very unstable inside buildings.

- Active Beacons

  This method has been employed since the beginning of aeronautics. Beacons can be detected very accurately, supplying very precise information about position. Furthermore, computing time is not expensive. The main problem is the expensiveness of installation and maintenance. An example of how beacons may help on localisation of mobile robots is available at [KV10].

- Global Positioning System

  Global positioning system is a technique for navigation in open spaces. It is made of several satellites that transmit coded signals using advanced trilateration methods. Sensors on earth are able to detect the time that the signal takes to get to them. By knowing the distance, they are able to calculate their latitude, longitude and altitude. An example of how Global Positioning System (GPS) can be used on mobile robots is available at [MK10].

- Positioning based on landmarks

Positioning based on landmarks is based in the detection of marks that help the robot to situate in the environment. An example of this situation can be found in [HOP11].

- Positioning based on maps

  Positioning based on maps also known as map matching is a technique where robots use their sensors to build a local map of the environment that is then used to match with a previously supplied global map of the environment.

- Positioning based on wireless networks (mobile networks and Wi-Fi)

  The popularity of this kind of positioning systems as increased over the last years. This technique consists on using the mobile network signal and/or Wi-Fi signal to increase the precision when determining the position of objects. For instance, in mobile phones, the "A-GPS" or "Assisted GPS" logo can be often found, which means that the mobile phone not only uses the GPS signal to determine its localization but also the mobile network signal. A paper comparing two methods to estimate the position of a mobile robot in an indoor environment using only odometric calculus and the WiFi energy received from the wireless communication infrastructure is presented in [OnPS06]. Another paper presenting a mobile robot that autonomously navigates in indoor environments using WiFi sensory data is available in [BV10].

1.1.6 Control Architectures

To ease the integration of the different modules that take part in a robotic system, control architectures were developed. They represent the way these modules are organised between them [DJ96]. There are different control architectures that are more, or less adequate to different applications and environments. Under a functional point of view, these architectures are currently grouped in three main categories: deliberative, reactive and hybrid.

Deliberative architectures, also known as hierarchical architectures, are based in the hypothesis of a system based on symbols [NS76] and are usually very well structured in a traditional bottom-up approach. The work of J.S. Albus [Alb92, Alb99] is based in this architecture. Reactive architectures, in contrast to deliberative ones, are very related to action. They are also called as behavior based architectures [Ark98, Bro85, MWDM98]. Hybrid architectures have some characteristics from both deliberative and reactive architectures. They provide very fast responses to changes in the environment as reactive architectures do, while achieving more complex tasks due to their higher computational and representation power that are a characteristic of deliberative architectures. It is possible to find examples of systems that use this architecture in [Ark86] and [Gat91].

Under a topological point of view, these architectures are grouped in horizontal and vertical. In the horizontal architectures, all levels have access to perception and determine the corresponding action. In vertical architectures, there is a perception level and another one of actuation. The sensed perceptions are processed in the different existing levels until an action is finally taken.

During the last twenty years, the theory of agents has become more popular. An agent can be defined as a piece of software conceived for reaching some goal. Agents encapsulate functionality, goals and intentions and they communicate with other agents to establish some kind of cooperation. They allow the creation of complex systems because one can create a distributed system made out of several agents. They are also easily expandable and changeable by adding, modifying or deleting one agent. More information about intelligent agents can be found in [WJ95].

1.1.7 Soft Computing and Robotics

SC refers to a collection of computational techniques in computer science, AI, Machine Learning (ML) and some engineering disciplines, which attempt to study, model, and analyse very complex phenomena: those for which more conventional methods have not yielded low cost, analytic, and complete solutions. SC differs from conventional (hard) computing in that, unlike hard computing, it is tolerant of imprecision, uncertainty, partial truth, and approximation. In effect, the role model for SC is the human mind. The guiding principle of SC is: Exploit the tolerance for imprecision, uncertainty, partial truth, and approximation to achieve tractability, robustness and low solution cost.

At this juncture, the principal constituents of SC are FL, ANN computing, Evolutionary Computing (EC), ML and Probabilistic Reasoning (PR), with the latter subsuming belief networks, chaos theory and parts of learning theory.

SC is nowadays broadly used in robotics in areas such as control systems, behavior arbitration, reinforcement learning, manipulation, collision avoidance and automatic design. More information can be found on books [JF98] and [ATKTJ00].

FL has been applied to robotics research. This is due to the easiness in dealing with the uncertainty and vagueness existing in the information given by sensors and the possibility of defining expert knowledge by means of "if-then" rules. Major works done in this area will be mentioned but more interested readers are referred to [Saf97].

- Design and coordination of behaviours: due to the possibility of dealing with uncertainty and vagueness using FL, it has been widely employed in the design of behaviours. An example of this feature is Shaphira's architecture [KM97] made of a set of fuzzy behaviours (coded with Mandani rules) capable of executing complex tasks. In this architecture, the level of belief of the rule preceding is used to compute its level of activation. Aguirre et al. [AG00a] adopted a very similar approach when defining some basic behaviours (follow-wall, navigate-in-walkway, avoid-obstacle, etc) used in fuzzy sets.

- Designing of maps: FL has been satisfactorily employed in the elaboration of maps, using both geometrical and topological approaches. Gasós and Saffiotti in [GS99], propose a technique for the elaboration of geometrical maps using sonar in which fuzzy borders are projected into a map of cells. In a similar way, Aguirre et al. [AG00b] present an hybrid approach (geometrical and topological) for the elaboration of maps using fuzzy segments originated from the information of ultrasonic sensors. In [GSFVGG97] and [OUV98] other approaches using fuzzy logic for the elaboration of maps are presented.

- Localization: FL has also been employed to compute the robot position using fuzzy measurements. Instead of representing the robot position using probabilistic measurements, some authors have used a possibility region to indicate the position of the robot. This approach allows the easier integration of the fuzzy maps previously mentioned. Some examples can be found at [GS99] and [SW96].

- Perception: FL has been used in the design of perceptual systems. Howard et al. propose in [HST01] a method based in FL to analyse the characteristics of a terrain in which a mobile robot is moving. They used a visual system for computing characteristics like transversability and discontinuity as fuzzy measures that were then used by the controller of the robot to compute the best strategy for the movement. Le et al. show in [LJW98] a fuzzy visual system to detect the borders of a road.

1.2 Human-Robot Interaction

HRI is the study of interactions between people (users) and robots. HRI is multidisciplinary with contributions from the fields of Human-Computer Interaction (HCI), AI, robotics, natural language understanding, and social science (psychology, cognitive science and anthropology).

The basic goal of HRI is to develop principles and algorithms to allow more natural and effective communication and interaction between humans and robots. Research areas range from how humans will work with remote, tele-operated machines to peer-to-peer collaboration with anthropomorphic robots.

Many scientists in this field study how humans collaborate, interact and use that information to motivate their research on how robots should interact with humans. As the goal of researchers it to make robots think, behave and react like an human does, psychology plays an important role in this area.

HRI has been a topic of both science fiction and academic speculation even before any robots existed. Because HRI usually depends on knowledge about human communication, many aspects of HRI are an extension of human communications topics that are much older than robotics.

In this work, efforts are centred in the area of HRI where robots are provided with some kind of intelligence that gives them the possibility of achieving human like behavior. The term “socially interactive robots” will be used to describe robots for which social interaction plays a key role and distinguish them from other robots that are based on “conventional” HRI, such as those used in tele-operation scenarios.

In the next section a brief explanation about social robots will be given followed in the succeeding sections by a description about different areas that are part of HRI.

1.2.1 Social Robots

A social robot can be viewed as an autonomous robot which is able to interact and communicate with humans by means of social behaviours and rules. In [HMW$^+$09] authors address several issues such as the meaning of “social robot”, the interdisciplinary research aspects of social robotics and how these different aspects are interlinked. They also argue that form, function, and context have to be taken systematically into account in order to develop a model to help us understand social robots.

From a long time ago researchers have been fascinated by the possibility of developing robots that could interact with other people and robots. In the 1940s Walter [HW] built a robot capable of interacting with another in a seemingly “social” manner, although there was no explicit communication or mutual recognition between them.

Some years later, Deneubourg and his collaborators pioneered the first experiments on stigmergy (indirect communication between individuals via modifications made to the shared environment) in simulated and physical “ant-like robots” [BHD94], [DGF$^+$90]. This idea is based on the concept of colonies of insects which are able to work for a common goal, although each individual works alone.

Similar principles can be found in multi-robot or distributed robotic systems research [Mat95]. Such societies are anonymous, homogeneous groups in which individuals do not matter. This type of “social behavior” has proven to be an attractive model for robotics, particularly because it enables groups of relatively simple robots to perform difficult tasks (e.g., soccer playing).

Many researchers in this area have focused on “benign” social behavior. This means that nowadays robots also play the role of assistants, companions or pets, in addition to the traditional role of servants.

Robots in individualised societies exhibit a wide range of social behaviours. In [Bre03], Breazeal defines four classes of social robots in terms of: (1) how well the robot can support the social model that is ascribed to it and (2) the complexity of the interaction scenario that can be supported. Those classes are defined as follows:

- Socially evocative. Robots that rely on the human tendency to anthropomorphize and capitalise on feelings evoked when humans nurture, care, or involved with their “creation”.

- Social interface. Robots that provide a “natural” interface by employing human-like social cues and communication modalities. Social behavior is only modelled at the interface, which usually results in shallow models of social cognition.

- Socially receptive. Robots that are socially passive but that can benefit from interaction (e.g. learning skills by imitation). Deeper models of human social competencies are required than with social interface robots.

- Sociable. Robots that pro-actively engage with humans in order to satisfy internal social aims (drives, emotions, etc.). These robots require deep models of social cognition.

Complementary to this list the following three classes can be added:

- Socially situated. Robots that are surrounded by a social environment that they perceive and react to ([DO02]). Socially situated robots must be able to distinguish between other social agents and various objects in the environment.

- Socially embedded. Robots that are: (a) situated in a social environment and interact with other agents and humans; (b) structurally coupled with their social environment; and (c) at least partially aware of human interactional structures (e.g., turn-taking) [DO02].

- Socially intelligent. Robots that show aspects of human style social intelligence, based on deep models of human cognition and social competence [Dau95], [Dau98].

Robots that interact with people have to exhibit some characteristics:

- to express and/or to perceive emotions;

- to communicate with high-level dialogue;

- to learn/recognise models of other agents;

- to establish/maintain social relationships;

- to use natural cues (gaze, gestures, etc.);

- to exhibit distinctive personality and character;

- may learn/develop social competencies

Social robots can be used in a wide range of applications like toys, educational tools, therapeutic aids, etc. A survey and taxonomy of current applications is given in [FND03]. As they play different roles (often operating as partners or assistants with different people) they have to show flexibility and adaptability.

Social robots have different shapes and functions ranging from robots whose sole purpose and only task is to engage people in social interactions (Kismet, Cog, etc.) to robots that are engineered to adhere to social norms in order to fulfil a range of tasks in human-inhabited environments (Pearl, Sage, etc.) [Bre02], [NBG^{+}99], [PMP^{+}03], [Sca01].

Some of these robots use deep models of human interaction and proactively encourage social interaction. Others show their social competence only in reaction to human behavior, relying on humans to attribute mental states and emotions to the robot [Dau97], [Duf03], [Pea01]. Regardless of the function, building a socially interactive robot requires considering the human in the loop: as designer, as observer, and as interaction partner.

Socially interactive robots are important for domains in which robots must exhibit peer-to-peer interaction skills, either because such skills are required for solving specific tasks, or because the primary function of the robot is to interact socially with people. A discussion of application domains, design spaces, and desirable social skills for robots is given in [Dau03], [Dau02].

One area where social interaction is desirable is “robot as persuasive machine” [Fog99], i.e., the robot is used to change the behavior, feelings or attitudes of humans. This is the case when robots mediate human-human interaction, as in autism therapy [WDOH01]. Another area is “robot as avatar” [PC98], in which the robot functions as a representation of the human. For example, if a robot is used for remote communication, it may need to act socially in order to effectively convey information.

In certain scenarios, it may be desirable for a robot to develop its interaction skills over time. For example, a pet robot that accompanies a child through his childhood may need to improve its skills in order to maintain the child interest. Learnt development of social (and other) skills is a primary concern of epigenetic robotics [DB99], [Zla01].

Some researchers design socially interactive robots simply to study embodied models of social behavior. For this use, the challenge is to build robots that have an intrinsic notion of sociality, that develop social skills and bond with people, and that can show empathy and true understanding. At present, such robots remain a distant goal [Dau97], [DB99], the achievement of which will require contributions from other research areas such as artificial life, developmental psychology and sociology [Res01].

Although socially interactive robots have already been used with success, much work remains to be done in order to increase their effectiveness. For example, in order to make socially interactive robots be accepted as “natural” interaction partners, they need more sophisticated social skills, such as the ability to recognise social context and convention.

Additionally, socially interactive robots will eventually need to support a wide range of users: different genders, different cultural and social backgrounds, different ages, etc. In many current applications, social robots engage only in short-term interaction (e.g., a museum tour) and can afford to treat all humans in the same manner. But, as soon as a robot becomes part of a person life, that robot will need to be able to treat him as a distinct individual [Dau98].

1.2.2 Human Oriented Perception

Perceiving the world as humans do is a desirable ability to achieve for social robots. In addition to the perception in conventional tasks such as navigation, localization, obstacle avoidance, they must be able to perceive things similarly to humans.

Robots also need human oriented perception. This means that they should be able to track people and human features such as bodies, faces, hands and others, to interpret human emotions including affective speech, discrete commands and natural language and to recognise facial expressions, gestures and other kind of human activity.

Once a robot is able to recognise and track the people in its vicinity, it should be able to detect their interest in establishing an interaction with it. In that task, several types of signals from the human can be taken into account (both verbal and non-verbal). Some authors [BFJ^{+}05a] use sound source localization or speech recognition besides visual perception to detect which persons are the most interested. In other cases, facial expressions [SKKB01], [CHF08] or hand gestures [GNS^{+}02], [PL11] are analysed. Finally, other authors [KC03] propose the use of non-verbal signals present in physiological monitoring systems that include skin conductance, heart rate, pupil dilation and brain and muscle neural activity.

People Detection and Tracking

In HRI, one of the basic tasks to solve has to do with the detection, identification and subsequent tracking of each of the interlocutors of the robot. First of all, the robot should identify which objects in its environment are its potential human interlocutors so it knows where the Region of Interest (ROI) which may provide most of the interaction cues is located. Then, it should be prepared to keep track of the history of each person, and "remember" what they did and what actions they took in the past. To do so, it has to be able to track the pre-detected people and to register their "actions".

For HRI, an important challenge is to find efficient methods for people tracking in the presence of occlusions, variable illumination, moving cameras, and varying background. There is extensive literature about this topic [Gav99], [LWT03], [SMC05] and [PAGSMS12]. In the next chapter, more information about this topic, which represents one of the main focus of research of this thesis, will be given.

Speech Recognition

Speech recognition (in many contexts also known as automatic speech recognition or computer speech recognition) is the process of converting a speech signal to a sequence of words, by means of an algorithm implemented as a computer program. In terms of technology, most of the technical text books nowadays emphasise the use of hidden Markov model as the underlying technology. The dynamic programming approach, the ANN based approach and the knowledge-based learning approach have been studied intensively in the 1980s and 1990s.

Speech recognition can be divided into detecting who is the speaker, what did he say and how did he say it [Bre02]. Depending on what information the robot requires, it may need to perform speaker tracking, dialogue management, or emotion analysis. Some applications of speech in robotics include [MAF$^+$99], [Oku01], [SAS01], [Bre02], [AST11].

Gesture Recognition

Humans usually use gestures while they communicate in order to clarify the speech. For instance, when a speaker wants to call others attention, he or she usually shakes his or her arm. Vision is the most adequate method to recognise gesture recognition as it does not force people to wear some specific hardware.

Gesture recognition is human interaction with a computer in which human gestures, usually hand motions, are recognised by the computer. Recognising gestures as input might make computers more accessible for the physically-impaired and make interaction more natural for young children. It could also provide a more expressive and nuanced communication with a computer.

In this work specific gestures or expressions are also recognised. Other methods can be found in [PSH97], [WH99], [PL11].

New HRI devices

Lately, in order to calculate depth information, others sensors such as the Kinect [Mic10] are being used in Human-Robot Interaction. The hardware that comprises the Kinect is different from the stereo camera used on this thesis. The Kinect features a RGB camera, a depth sensor and a multi-array microphone running proprietary software. Instead of computing depth from a pair of stereo color images it uses an infrared projector and an infrared camera which are able to compute depth. The IR camera and the IR projector form a stereo pair with a baseline of approximately 7.5 cm. The IR projector sends out a fixed pattern of light and dark speckles. The pattern is generated from a set of diffraction gratings, with special care to lessen the effect of zero-order propagation of a centre bright dot [Pri05]. Depth is calculated by triangulation against a known pattern from the projector. More information is available in [Zha12].

Several tools as the openNI SDK and the NITE middleware [ope10] have been developed to manage the depth, color, infrared and audio information received from the hardware device. These tools allow to perform functions such as hand location and tracking; a scene analyser (separation of users from background); accurate user skeleton joint tracking; various gestures recognition, and so on. Although the way of operating is different comparing to a stereo camera, most of the algorithms presented in this thesis could be adapted to this sensor and implemented in future works.

Due to the low cost of the device and the availability of tools to manage it, it is more and more popular in the scientific field and several works have been published, namely in the field of HRI. Among those works that use the Kinect sensor, we will mention some which are somehow related with this thesis. Some works [CV12], [FLW$^+$12], [BJ11] try to solve robot navigation and obstacle avoidance issues using fuzzy logic based controllers. Also, it is used to build complete 3D models using soft-computing [VGRC12]. This method computes the movement performed by a mobile robot by means of a 3D models registration algorithm.

Facial Perception

Face detection and recognition is also important to identify possible people in the environment that wish to interact with the robot. Furthermore, gaze direction and head pose are valuable pieces of information about the real interest of people in communicating with the robot. Another important feature is face expression, that can tell the robot about the emotional state of the individual. In [LS00] basic approaches for facial expression recognition are discussed.

In [LKH$^+$03] a multi-modal attention system is shown. This approach uses a pan-tilt camera for face recognition, two microphones for sound source localization and a laser range finder for leg detection. Shifting attention is carried out by turning the camera towards the person which is currently speaking. In [BFJ$^+$05b], the authors present a system that makes use of visual perception, sound source localization and speech recognition to detect, track and involve people into interaction. In [BFJ$^+$05b] the goal is that the robot interacts with multiple persons without focusing its attention on a single person.

1.2.3 User Modeling

For robots to interact with people in a human-like manner, they have to perceive human social behavior [Bre02]. To achieve this task they should detect and recognise human action and communication as well as interpret and react to their behavior. This is called user modeling.

User modeling can be quantitative, based on the evaluation of parameters or metrics. The stereotype approach, for example, classifies users into different subgroups (stereotypes), based on the measurement of pre-defined features for each subgroup [Lor95]. User modeling may also be qualitative in nature. Interactional structure analysis, story and script based matching, all identify subjective aspects of behavior.

User models usually describe a user or group of users. These models can be static (previously defined) or dynamic (adapted or learnt). Information about users could be done explicitly (direct questioning) or implicitly (inferred through observation).

User models help the robot understand human behavior and dialogue. They also shape and control feedback to users and are useful for adapting the robot behavior to accommodate users with varying skills, experience, and knowledge.

Fong et al. [FTB01] employ a stereotype user model to adapt human-robot dialogue and robot behavior to different users. Schulte et al. [SRT99] describe a memory-based learner used by a tour robot to improve its ability to interact with different people. Koo et al. [KPKK11] propose a dual-layer user model to generate descriptive service recommendations for user-adaptive service robots.

1.2.4 Intentionality

Some authors argue that humans use three strategies to understand and predict behavior. The prediction based on the physical characteristics of the object, the prediction based on the design and functionality of the artifacts and the intentional stance that assumes that the system actions result from its beliefs and desires.

For the robot to be able to interact socially, it should demonstrate that it is intentional. For example, a robot could demonstrate goal-directed behaviours, or it could exhibit the attentional capacity. If it does so, then the human will consider the robot to act in a rational manner.

Humans use a variety of physical social cues to indicate which object is currently under consideration. Scassellati [Sca03] doted its robot Cog of gaze following, imperative pointing, and declarative pointing capacities. Park et al [PL11] also present a real-time 3D pointing gesture recognition algorithm for mobile robots, based on a cascade hidden Markov model (HMM) and a particle filter. Kopp and Gardenfors [KG01] consider that attentional capacity is a fundamental requirement for intentionality. After identifying the relevant objects in the scene, the robot should focus its attention at one of them and direct its sensors to it. Marom and Hayes [MH99], [MH01] consider attention to be a collection of mechanisms that determine the significance of stimuli. Their research emphasises the development of pre-learning attentional mechanisms, which help to reduce the amount of information that an individual has to deal with.

Kozima and Yano [KY01a], [KY01b] support the theory that a robot must have goal-directed behavior to be intentional. Breazeal and Scassellati [BS99] describe how Kismet transmits intentionality through motor actions and facial expressions. Schulte et al. [SRT99] discuss how a caricatured human face and simple emotion expression can transmit intention during spontaneous short-term interaction. For example, a tour guide robot might have the intention of making progress while giving a tour. Its facial expression and recorded speech playback can communicate this information.

1.2.5 Soft Computing Techniques for Human Robot Interaction

Soft-Computing techniques have also been used in HRI. As discussed before, HRI is an area where robots should be able to "think" like humans, to learn new processes and to adapt themselves to the people in their surroundings. Therefore, SC techniques are important in order to improve many of these characteristics. In [BS03] several SC techniques are applied to service robotic systems for comfortable interaction and safe operation.

Genetic computation can be used as a powerful exploratory method to acquire knowledge, with reduced efforts in human design. This approach is nowadays one of the most popular ones: the simple biological metaphor at the base of genetic processing has been widely acknowledged as an efficient engine to design reactive, adaptive and evolving agents. The assumption that GA are a promising approach to be used for modeling adaptive systems has been successfully verified in many experiments and in different areas. Schultz presents a method for learning robot behaviours using GA in [Sch94]. Wang Yan-ping and Wu Bing [YpB10] propose a method of mobile robot path planning based on modified genetic algorithms to achieve its goals in dynamic environments and to avoid obstacles.

ANN have also been employed to optimise agent behavior. The impact of multi-agent strategies, often associated to evolutionary behavior have shown to be beneficial for the transfer of knowledge across multiple functions and for a successful learning of the systems. In [BBB$^+$98], Boehme et al. use ANN for gesture-based interaction between a mobile robot and its user. Shuang et al. [LDX11] propose a hybrid recognition algorithm based on the combination of rough set theory and ANN in order that a mobile robot is able to recognise the shape of objects in dynamic surroundings.

FL is used as an inference engine for complex distributed, inference-based applications. Here fuzziness is usually used to represent and reason about vague knowledge with fuzzy productions rules. FL also "imitates" human reasoning and could be used to give robots an human similar way of "thinking". FL can also be used for recognising facial emotional expression and for coordinating bio-signals with robotic motions. In [KC03], several sets of fuzzy rules are used for estimating intent based on physiological signals. In [ETNMT11], El-Teleity presents a control strategy in which four different reactive behaviours are combined by means of a fuzzy supervisor that controls the movement of an autonomous robot.

1.3 Related Work

1.3.1 People Detection and Tracking

Although people detection and tracking with a single camera is a well explored topic, the use of stereo technology for this purpose concentrates an important interest. The proposal hereby described is intended to stereo vision but it could be easily adapted to use other kinds of vision and depth sensors. The availability of commercial hardware to solve the low-level problems of stereo processing, as well as the lower prices for this type of devices, turn them into an appealing sensor to develop intelligent systems. SV provides a type of information that brings several advantages when developing human-machine applications. For instance, some advantages of applying SV to object tracking, can be found in the work of [GBL$^+$11] where an example of a real-time tracking algorithm for following a 3D position of a generic spatial object is presented. In the work of [YN12] it is possible to find an application of SV to agriculture, although it focuses only on people detection rather than the tracking. Another application is presented in the work of [SELG10] which describes a 3D detection and tracking of pedestrians in urban traffic scenes. The system is built around a probabilistic environment model which fuses evidence from dense 3D reconstruction and image-based pedestrian detection into a consistent interpretation of the observed scene, and a multi-hypothesis tracker to reconstruct the pedestrians' trajectories in 3D coordinates over time.

The possibility to know the distance from the camera to the person is an advantage of SV over other techniques. This could be of great assistance for tracking as well as for a better analysis of his/her gestures. To achieve so, stereo correlation algorithms are used. They are able to match the pixels of several (two or more) different adjacent cameras and compute the distance of those pixels to the camera. On this work, it is used the stereo correlation algorithm of the camera manufacturer ([Res10]). Nevertheless, many algorithms exist namely the recent ones of [AK10] and [Zic12] that aim to improve the accuracy of this kind of algorithms. There is also a less exploited field on stereo correlation algorithms which use several (at least two) PTZ (Pan-Tilt-Zoom) cameras. Stereo vision using dual-PTZ-camera system, compared with using dual-static-camera system, is much more challenging. In [WZ08] authors propose a novel stereo rectification method for dual-PTZ-camera system, which is essential to greatly increase the efficiency of stereo matching.

Another positive point of SV over monocular vision is that information regarding disparities becomes more invariable to illumination changes than the provided by a single camera. There are authors that have studies the problem of illumination in stereo computing as the work of [NG10] which proposes a new illumination-invariant dissimilarity measure in order to substitute the established intensity-based ones. Robustness to luminance changes is a very advantageous factor for the development of background estimation techniques [LHH12], [DDCF01], [HGW01], [EKB98]. As a basis for improved people detection, a correct segmentation of the image is also desirable. To achieve that goal, a good scene calibration enables the system to process the input video in a different way depending on the camera position and the scene characteristics. In the paper of [PATF13], an automatic method to calibrate the scene, for detecting and tracking people systems, is presented based on measurements of video sequences captured from a stationary camera.

In the majority of works, one or several cameras, often placed in elevated positions, are used [Har04, GK04]. In [Har04] a method for locating and tracking people in stereo images is presented using occupancy maps. Before the people detection process takes place, an image of the environment is created through a sophisticated image analysis method. Once the background image is created, the objects that do not belong to it are easily isolated, and an occupancy map and a height map are built. The information from both maps is merged to detect people using simple heuristics. People tracking is performed by using a Kalman filter combined with deformable templates. In their work, a stereoscopic system is used which is located three meters above the ground, on a fixed position. In the work of Grest and Koch ([GK04]) a Particle Filter (PF) ([IB98]) is used to estimate the position of the person and to create colour histograms of the face and the chest regions of one person and the SV is used to compute its real position. However, stereo and colour were not integrated in the tracking process and they use cameras positioned in different parts of a room rather than only one stereo camera. A more recent work, the one from [SM11], proposes a method to locate and track people by combining evidence from multiple cameras using the homography constraint. The algorithm computes the amount of support that basically corresponds to the "foreground mass" above each pixel. Therefore, pixels that correspond to ground points have more support. The support is normalised to compensate for perspective effects and accumulated on the reference plane for all camera views. Two other recent works, the one from [CHX$^+$10] and the one from [AD12] only use a single camera. Nevertheless their camera is also placed at a higher than a human head position in order to have a more clear field of view.

However, to privilege interaction with HRI, position of the camera should usually be placed lower than the height of the person, like in [DGHW00]. Here authors present a system capable of detecting and tracking several people. Their work is based on a skin detector, a face detector and the disparity map provided by a stereo camera. Besides improving the visibility of the face and arms of the person, these methods are more adequate for their implementation in robotic systems that require interaction with people. Studies carried out, show that in order to improve the acceptance of robots by humans, it is important that they are located in a lower position than the later [FND03]. Otherwise the person could feel intimidated.

When tracking people, there might occur occlusion situations that affect the performance of trackers. On multi-camera approaches, this problem should happen less than when using a single camera. There are authors that have done some work on this problematic, such as [HH12], which circumvents this problem by performing tracking based on observations from multiple wide-baseline cameras. However, [CDL10] presents an approach for tracking multiple persons on a mobile robot with a combination of colour and thermal vision sensors, using several new techniques and a single camera. In their approach an algorithm for detecting occlusions is introduced, using a machine learning classifier for pairwise comparison of persons (classifying which one is in front of the other). On our work, we also deal with this problem using only one camera, with the help of a occlusion map.

The Kalman/mean-shift (described in [CR00]) is employed in different tracking approaches. In their work, Comaniciu and Ramesh combine the well known mean-shift algorithm with colour information to locally move the search region towards the gradient direction of the Bhattacharyya coefficient described in [ATR97]. The Kalman filter is employed to predict the position of the target in the next frame. Another colour-based particle filtering technique that uses this kind of information is the one described by [NKMG03], where each particle represents a possible position and size of the tracked object. [SB10] also present a method where they combine the use of Monte Carlo sequential filtering for tracking and Dezert Smarandache theory (DSmT) to integrate the information provided by different colour and position cues. [MTACS02] present a system able to detect and track a single head using the Kalman filter. They combine colour and stereo information but head colour does not provide enough information to distinguish among different users. In [Har04] and [MSAGS07], authors present an approach to detect and track several people using *plan-view maps*. They use information provided by an *occupancy map* and a *height map* using the Kalman filter. More recent works such as the one from [MJK12] proposes a novel and efficient method of tracking, which performs well even when the target takes a sudden turn during its motion. Their method arbitrates between KF and Optical flow (OF) to improve the tracking performance and uses a laser sensor to measure distance. On the other hand [FSA11] uses the classic Kalman filter for tracking and uses a low-level recognition system to properly distinguish among the targets.

When merging different unreliable or imprecise sources of information one can choose between using probabilistic/mathematical based models ([MSMCMCCP09], [LS09], [KLM10]) or SC techniques. An example of a SC technique based on FL can be found in [HLK09]. They decompose the input-output characteristics into noise-free part and probabilistic noise part and identify them simultaneously. Other SC techniques applied to computer vision have already been used in different works namely the ones from [KjB97] and [Blo08]. [MB10] presents a pattern classifier system for the detection of people using laser range finders data is presented. The approach is based on the quantified fuzzy temporal rules (QFTRs) knowledge representation and reasoning paradigm, that is able to analyse the spatio-temporal patterns that are associated to people. More information and work on this subject can be found in [SCFERB$^+$09], [SCSD09] and [NKDdW09]. In the current work, FL ([YF94]) is privileged in order to have the possibility of dealing with uncertainty and vagueness in a flexible manner as well as to avoid restrictions when representing imprecision and uncertainty with probabilistic models. Regarding object detection, different works, as the one from [IBP06], are supported by FL approaches.

PF are widely used on object tracking algorithms. They can estimate the state of a dynamic system $x(t)$ from sequential observations $z(t)$ as refereed in different works as the ones from [GS95]), [IB98] and [Kit96]. They are able to manage multiple hypotheses simultaneously, by dealing naturally with systems where both the posterior density and the observation density are non-Gaussian. However, they may present some problems when used for multi-target tracking. Firstly, the standard version of the PF, does not define a method to identify individual targets. Furthermore, particles generated by this kind of filter quickly converge to a single target, discarding the rest of them (also known as the coalescence or particle “hijacking” problem). Another problem is that it can suffer from exponential complexity as the number of targets increases. [VGP05] as well as [KBD05] and [OTdF$^+$04] propose different approaches to deal with these problems. A Multi Particle Filter (MPF) consists of employing an independent PF for each target and an interaction factor which modifies the weight of particles in order to avoid the coalescence problem. A more recent work, the one of [MS12], presents a novel approach, based on drift homotopy for stochastic differential equations, for improving particle filters for multi-target tracking. Drift homotopy is used to design a Markov Chain Monte Carlo step which is appended to the particle filter and aims to bring the particle filter samples closer to the observations while at the same time respecting the target dynamics. Another approach which is aimed for people tracking is the one from [PMC12] in which a generic online multi-target track-before-detect (MT-TBD) that is applicable on confidence maps used as observations is proposed. The main novelty is the inclusion of the target ID into the particle state, enabling the algorithm to deal with unknown and large number of targets.

Another way of dealing with uncertainty and vagueness issues susceptible of being found in PF are the so called Fuzzy Logic based Particle Filter (FLPF), which is a concept that has been applied by some authors. In [ZZ08], the ideal number of generated particles is computed using a Fuzzy System(s) (FS). In [YJJMMT07], a fuzzy adaptive PF for the localization of a mobile robot is proposed, whose basic idea is to generate samples at high-likelihood using a FL approach. [SMYF10] present a particle filtering approach in which particles are weighted using a fuzzy based color model for object which discriminates between background and foreground elements. In their approach, only that information is fuzzified and used to evaluate the particle. [KB05] present a FLPF algorithm for tracking a manoeuvring target. In their work, the nonlinear system which is comprised of two-input and single-output are represented by fuzzy relational equations. In [ZB09] a PF approach, where face detection information is also used to enhance the performance of the PF, is described. In the current approach, the problem of merging different information sources, usually accompanied by vagueness and uncertainty, is solved by using a new approach based on a PF which generates particles that are evaluated by means of FL.

We finish this section by citing some papers that use the Kinect sensor which, as said before, is a sensor that is becoming popular in this field of study, for solving people detection and/or tracking problems. Albiol et al [AAOM12] propose the concept of bodyprints to perform re-identification of people in surveillance videos. The Kinect is placed in a high position so that several people are seen from above. They create a database of 40 people and argue that their bodyprint concept is very robust to changes of pose, point of view and illumination. Also, the algorithm could be used for tracking people using networks of non-overlapping cameras. Another proposal [LSA10] combines a multi-cue person detector for RGB-D data with an on-line detector that learns individual target models. It neither relies on background learning nor has a ground plane assumption. It uses 3 Kinect devices mounted vertically and target appearance models must be learnt. Others authors [RBMHMU12] present an approach for people detection and tracking by an autonomous mobile robot, using the Kinect (although they claim that their methodology can be applied to any RGB-D system). They use 2D space information to detect features of people, like face and skin, and 3D information to segment the people silhouette. Some authors [SAJ$^+$13] claim that the Kinect presents some limitations for its use on a mobile platform and propose to add to the system a thermical sensor (thermopile) mounted on top of a mobile platform. They propose the implementation of an evolutionary selection of sequences of image transformation to detect people through supervised classifiers and show that the people detection error is reduced. Finally, a method which uses the Kinect to achieve human object recognition using the depth and color information of the shirt a person is wearing was proposed [SF13].

1.3.2 Human Response

In the fields of HCI and HRI, a main goal is to be able to replicate the behavior among humans by designing a system capable of interacting with humans in a natural way. There is certainly a long way to be traversed but there are interesting works in the literature about this topic.

One of the works [AKK08] is aimed for users that are located next to a screen. Specific cues such as head pose and eye gaze that are only detectable at very close distances were considered. Other authors [UPSP10] use different types of sensors to segment a human region of interest and to track his/her motion. They do not use colour information and show their results for only one person. Other examples are mostly based on face information [SBOGP08, SM04], focus orientation [HCPW03] or eye gaze [MI]. Most of these examples require that users are placed at relatively close distances from the camera and its reliability often depends on the level of his or her head motion. It is usual to find the use of specific face features [AS11], where a face and head gesture detector in video streams is used. The detector is based on face landmark paradigm in which appearance and configuration information of landmarks are used. Also machine learning is considered and [CHF08] propose a hybrid-boost learning algorithm for multi-pose face detection and facial expression recognition. Another option is to use algorithms based on extracted points from the subjects faces as well as their physiological responses [BPM$^+$08].

Different sources of information could be considered. For example, it is possible to consider sound source localization or speech recognition combined with face detection in order to study the human behavior [BFJ$^+$05a]. In other cases [SKKB01], the recognition of specific facial (lips) expressions allows to a robotic arm to operate autonomously via visual feedback. Most of the mentioned works require users to be close to the sensors and, also, they centre their study in only a few features.

In order to detect certain arms interaction gestures, we opted to use a fast but precise algorithm [AGSG$^+$07] where stereo information is also used to detect simple interest demanding gestures with the arms. Thus, in our proposal, we try to integrate different visual features detectable at several ranges of distances (close, medium and far) using a Hierarchical Fuzzy System(s) (HFS). It makes it a more robust and flexible Human Robot Interaction oriented system, at different distances. Our system is composed by a single stereo camera (similar to human vision) and avoids the use of specific “not human alike” hardware. Also, it is modular and it is possible to easily include new feature detection algorithms in future works.

Our proposal requires to fuse different information. The fusion of information has the problem that data is often affected by errors (which are linked, for instance, to unpredictable situations and to the physical specifications of the sensor). In order to handle these issues, soft computing techniques, as FL, can be used.

In the fields of HCI and HRI, FL has been used in several works. One option is to use FL clustering techniques to model an operator’s attention (based on eye gaze) and to develop a computational model for the attention and its allocation [LZW$^+$09]. As authors claim, their model can be limitative by only using one type of cue. Regarding emotion detection, in [MA07] a FL model transforms four physiological signals into arousal and valence and a second FL model transforms arousal and valence into five emotional states relevant to computer game play: boredom, challenge, excitement, frustration, and fun. Their approach makes use of several hardware devices that must be applied on the user to extract features which will be used as the fuzzy sets inputs. Although it is an interesting approach, the use of “human intrusive” sensors could originate a certain feeling of rejection respect to the system. In [ISH12], a new method for Emotion Recognition from Facial Expression using Fuzzy Inference System (FIS) is proposed. Their method is able to recognise emotions from partially occluded facial images. Another proposal [XLC08] is based on the Fourier transform to represent one facial expression and then to process the information using the fuzzy C-means algorithm to generate a spatio-temporal model for each expression type. These methods rely only on facial features which are hard to detect at higher distances.

In our proposal, one different feature extracting method is applied depending on the distance. Then, this sensorial information is fused using a HFS, taking into account its level of confidence. HFS [Tor02] allow the organization of several FS according to the type of information they cope with. In this work, thanks to the proposed HFS, sensorial information is handled according to the distance at which the tracked person is placed. Then, it is possible to compute his or her instantaneous, accumulated and average level of human response, for a specific period of time.

Finally, and before concluding this section, some works using the relative new Kinect depth sensor, and that are somehow related with the detection of human behavior, namely gestures detection, are presented. In [MBMM13] there is a proposal using an autonomous system for real-time human action recognition based on 3D motion flow estimation. They exploit colored point cloud data acquired with a Kinect sensor and summarise the motion information by means of a 3D grid-based descriptor. In [LYTZ13] authors present a novel vision-based markerless hand pose estimation scheme based on depth image sequences. The proposed scheme exploits both temporal constraints and spatial features of the input sequence, and focuses on hand parsing and 3D fingertip localization for hand pose estimation. Respect to body pose estimation, a work [OKO$^+$12] compares the Kinect pose estimation (skeletonization) with more established techniques for pose estimation from motion capture data, examining the accuracy of joint localization and robustness of pose estimation with respect to the orientation and occlusions. Experimental results present pose estimation accuracy rates and corresponding error bounds for the Kinect system. Finally, there is an application of pose estimation [MM13] in which a novel fall detection system based on the Kinect sensor is presented. The system is capable of detecting in real time walking falls. It is performed in accurate and robust way and without taking into account any false positive activities (i.e. lying on the floor). Velocity and inactivity calculations are performed to decide whether a fall has occurred.

Chapter 2

System and Techniques Description

In this chapter a description of the used hardware is firstly given. Then, in the second section, the basis of the used stereo algorithm will be presented followed by the colour modeling method. These two methods are crucial to all subsequent developed algorithms which are based in this kind of visual information. In the third section of this chapter, the basis of Principal Component Analysis (PCA) and Support Vector Machines (SVM), which correspond to those learning machine techniques that were several times employed, is described. Finally, Fuzzy Logic (FL) is presented, as this reasoning method is largely privileged in this thesis.

2.1 Hardware Description

The hardware system used in the different works presented in this thesis was comprised of a PeopleBot mobile robot ([Rob]), a stereoscopic system with a binocular camera [Res05] and a laptop for processing all the data. The camera and the laptop were mounted on the top of the robot structure as seen in Fig. 2.1. The used stereoscopic system allows the extraction of not only colour but also depth information. In our experiments sequences were recorded with a resolution of 320 x 240 pixels size at a 15 fps frame rate. At first, the used laptop was comprised of an Intel Pentium IV Central Processing Unit (CPU) working at 3.2 Ghz. Lately, the processor of the laptop was an Intel i5 CPU working at 2.67 GHz.

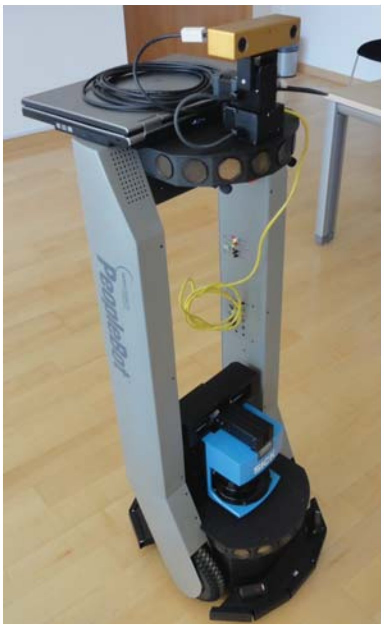


Figure 2.1: Peoplebot robot with a laptop and the Bumblebee stereo camera on the top.

The main sensor used in the current work, the stereo camera, captures two images from slightly different positions (calibrated stereo pair) which are transferred to the computer to calculate a disparity image containing the points matched in both images (see Section 2.2.1).

Although the moving functionality of the robot is not directly used in the presented works, the goal is to integrate the algorithms hereby presented with other works developed by different research teams of our group. The height of this robot is similar to the average height of a 8 to 10 years old child. We believe this is an advantage as it may favour the interaction of people with a social robot. As a matter of fact, people will feel more comfortable interacting with a robot as high as an human being and having its vision system placed on the top of it (simulating its eyes), as an human being does. This is one of the conditions that were prioritised in the different works as it is important for the research group to simulate the same conditions taking place on a Human to Human interaction. The system is aimed to be used in different social activities in which several people could participate while freely moving and interacting between a distance which varies from 0.5 to 5 meters (limitation due to the image resolution of the camera, angle of vision and real time performance restrictions).

2.2 Computer Vision

In this Section, a description of the two main computer vision techniques used in the different works is given. Firstly, stereo vision is introduced and then color modelling is described.

2.2.1 Stereo Modeling

In this section, the notions of stereo computation, which is the base of the entire system (people detection, tracking and human behaviour analysis), are presented. For a more detailed review, the interested reader is referred to [BBH03]. Please note that it is not our purpose to develop or present a new stereo matching algorithm. Instead of it, the software that comes with the camera ([Res10]) is used. The camera software already deals with lens distortion when performing stereo computation. It supplies an assembly optimized fast-correlation stereo core that performs fast Sum of Absolute Differences (SAD) stereo correlation. This method is known for its speed, simplicity and robustness, and generates dense disparity images.

The minimal possible Stereo Vision (SV) system is composed by a pair of cameras whose optical centres (O_l and O_r) are separated by a distance b. Let us assume, in order to simplify the explanation, that both cameras have identical optical characteristics and have coplanar vision planes (as in Fig.2.2). A SV system is able to capture two images (I_l and I_r) at the same instant. Both cameras are calibrated and the captured images are rectified in order to remove the deformations caused by lens distortion. Usually, the centre of one of the cameras is employed as reference system. In the present case, it will be the centre of the right image (named *reference camera image*).

A point $P = (X, Y, Z)$ in space projects to two locations ($p = (x, y)$ and $p' = (x', y')$) at the same epipolar line on each rectified images. The displacement of the projection in one image in comparison to the other is named disparity and the set of all disparities between two images is the so called disparity map. Disparities can only be computed for these points that are registered on both images but it is difficult to do it when there are occlusions or insufficient texture. The points whose disparity cannot be calculated are named *unmatched* points.

Knowing the intrinsic parameters of the SV system, such as the focal length (in the presented system this value is 6 mm), it is possible to reconstruct the three-dimensional structure corresponding to the disparity map.

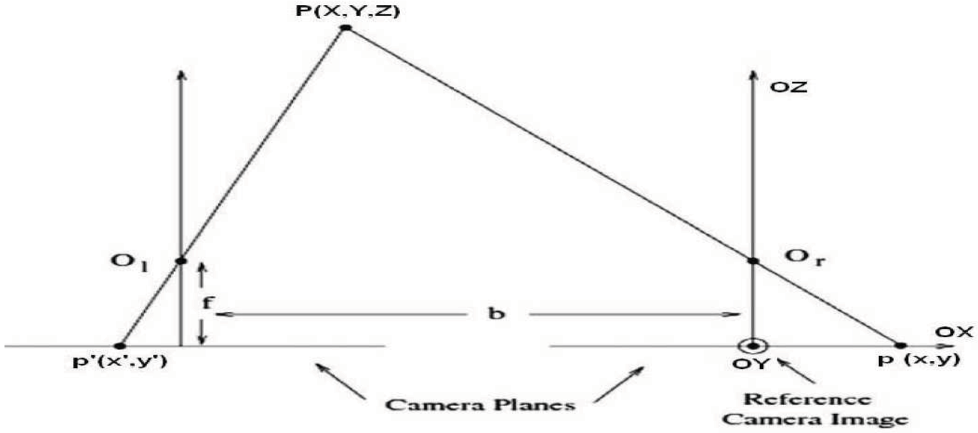


Figure 2.2: Minimal stereo system composed by two cameras.

In Fig.2.3 there is an example of a scene captured with a SV system. While in Fig.2.3a the left camera image I_l is shown and in Fig.2.3b the right camera image I_r (defined as the reference image) is shown, in Fig.2.3c it is possible to see the distance image I_z. In this image, brighter pixels indicate lower values of Z while darker ones represent farther distances. Black pixels represent unmatched points. The disparity map for this frame would be an image similar to Fig.2.3c, where lighter pixels would mean greater disparity and darker ones would represent smaller disparity. In addition, it is also important to take into consideration that distance information obtained from a stereo pair is affected by typical stereo errors, i.e., in the calibration, quantization and matching processes, as explained in ([MMN89] and [RA90]). Algorithms employing stereo information must properly deal with these errors

Although distance (stereo) information is used to improve the accuracy of the tracking algorithm, the tracking is done in the reference image, in the 2D domain, where the position of a person is the position of the centre of his/her face, which was originally detected by a face detector in the reference image. Thus, the position of a person is a (x_p, y_p) pair corresponding to a pixel within the reference image.

Figure 2.3: (a) Image of the left camera I_l captured with the Stereo Vision system. (b) Image of the right camera I_r (reference image). (c) Image of distance I_z.

2.2.2 Colour Modeling

Tracking objects using colour information is a well known problem which has been studied using different approaches [Bir98], [CR00], [GK04], [NKMG03].

The most frequently used method consists of using a histogram to represent a colour model $\hat{q}$ where each bin represents a colour region. As HSV colour space ([FvD82]) is relatively invariable to illumination changes, it has become a popular approach in this domain. A colour histogram $\hat{q}$ comprises $n_h n_s$ bins for the hue and saturation. However, chromatic information cannot be considered reliable when the value component is too small or too big. Therefore, pixels in this situation are not used to describe the chromaticity. Due to the fact that these pixels might have important information, the histogram includes also n_v bins to capture its luminance information. Thus, the resulting histogram is composed by $m = n_h n_s + n_v$ bins.

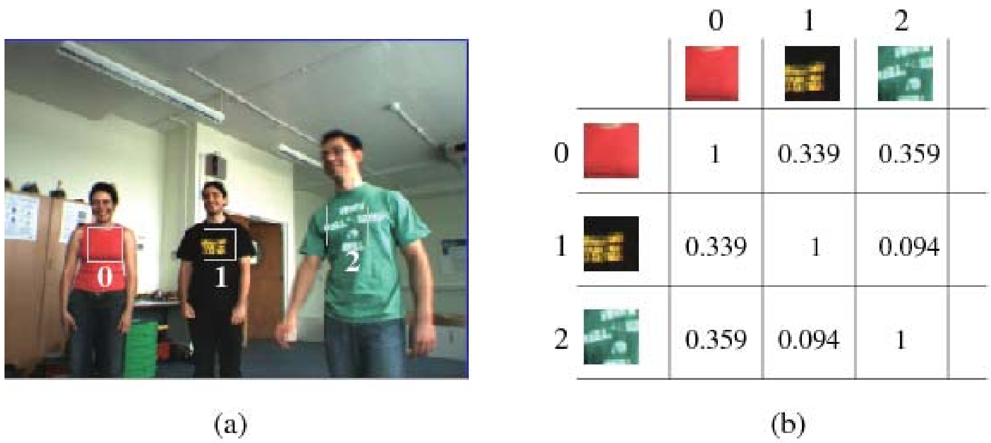
As stated in [Bir98], [CR00] and [NKMG03], an elliptical region of the image is used to create the colour model whose horizontal and vertical axis are h_x and h_y respectively. Let p_c be the ellipse centre and $\{p_j\}_{j=1,\ldots,n}$ the locations of the interior pixels of the ellipse. Let us also define a function $b : \Re^2 \rightarrow 1, ..., m$ which associates to the pixel at location p_j the index $b(p_j)$ of the histogram bin corresponding to the colour u of that pixel. It is now possible to compute the colour density distribution $\hat{q}$ for each elliptical region with:

$$\hat{q}(u) = \frac{1}{n} \sum_{j=1}^{n} k[b(p_j) - u], \tag{2.1}$$

where the parameter k is the Kronecker delta function. Please notice that the resulting histogram is normalised, i.e., $\sum_{u=1}^{m} \hat{q}(u) = 1$.

After calculating the colour model $\hat{q}$, it is possible to compare it with another colour model $\hat{q}'$ using the Bhattacharyya coefficient as described in [ATR97] and [Kai67]. In the case of a discrete distribution it can be expressed as indicated in Eq. 2.2. The result expresses the similarity between two colour models in the range of $[0, 1]$ where 1 means that they are identical and 0 means that they are completely different. An important feature of ρ is that both colour models, $\hat{q}$ and $\hat{q}'$, can be compared even if they have been created using regions of different sizes. In Fig. 2.4 there is an example of a frame taken from a video on the left and a table comparing different Region of Interest (ROI) with their corresponding Bhattacharyya coefficient on the right.

$$\rho(\hat{q}, \hat{q}') = \sum_{u=1}^{m} \sqrt{\hat{q}(u)\hat{q}'(u)}. \tag{2.2}$$



	0	1	2
0	1	0.339	0.359
1	0.339	1	0.094
2	0.359	0.094	1

(a) (b)

Figure 2.4: (a) Scene with three objects in it. (b) Bhattacharyya values for different images.

2.3 Machine Learning

As in the works presented in this thesis, different machine learning techniques were used, we introduce in this section two of those techniques: Principal Component Analysis and Support Vector Machines.

2.3.1 Principal Components Analysis

PCA [HD89] is a technique widely employed for dimensionality reduction. When PCA is used, an image is transformed into its principal components, i.e., those that contain the “most important” aspects of the data. PCA has the distinction of being the optimal linear transformation for keeping the subspace that has largest variance. It allows the identification of patterns in data, and to express the data in such a way as to highlight their similarities and differences. PCA is also used to compress data as, once these patterns have been found in the data, the number of dimensions may be reduced, selecting the percentage of information that is lost. The mathematical basis of PCA can be easily found in bibliography or on the Internet [Sim13].

A PCA projection represents a data set in terms of the orthonormal eigenvectors of the data set’s covariance matrix. A covariance matrix captures the correlation between variables in a data set. PCA finds the orthonormal eigenvectors of the covariance matrix as the basis for the transformed feature space. Eigenvectors can be thought of as the "natural basis" for a given multi-dimensional data set. Higher eigenvalues in the covariance matrix indicate lower correlation between the features in the data set. PCA projections seek uncorrelated variables.

Every data set has principle components, but PCA works best if data are Gaussian-distributed. For high dimensional data the Central Limit theorem allows us to assume Gaussian distributions.

Let us begin by calculating the variance of a single variable x as:

$$\sigma^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{X})^2}{n}$$

Then it is possible to calculate the variance of two variables, x and y, as:

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(x_i - \bar{X})(y_i - \bar{Y})}{n}$$

With the covariance, one is able to see how two variables vary together:

- In the case that covariance between two variables is positive, if one variable increases, the other will also increase.
- In the case that covariance between two variables is negative, if one variable increases, the other will decrease.
- In the case that covariance between two variables is zero, then both variables are completely independent of each other.

For a set of variables $< X_1, ..., X_n >$, (for instance, the features of a data set) it is possible to construct a matrix which represents the covariance between each pair of variables X_i and X_j where i and j are indexes of the feature vector.

$$cov(X) = \begin{bmatrix} var(X_1) & cov(X_1, X_2) & \cdots & cov(X_1, X_n) \\ cov(X_2, X_1) & var(X_2) & \cdots & cov(X_2, X_n) \\ \vdots & \vdots & \ddots & \vdots \\ cov(X_n, X_1) & cov(X_n, X_2) & \cdots & var(X_n) \end{bmatrix}$$

In this matrix the diagonal simply represents the variance of an individual variable. This matrix is symmetric, which means that, $cov(X_i, X_j) = cov(X_j, X_i)$.

Before using the concept of covariance in PCA, there is a first step which consists of subtracting the means $\bar{X}_i$ from each x_i before constructing the covariance matrix so that each $\bar{X}_i$ has a mean of zero. By subtracting the mean it is possible to rewrite the covariance matrix as the following matrix multiplication:

$$\mathbf{\Sigma} = \mathbf{\frac{1}{n} X X^T}$$

Then, by applying the spectral decomposition theory, we can factor the matrix above into:

$$\mathbf{\Sigma} = \mathbf{U \Lambda U^T}$$

where $\mathbf{\Lambda} = diag(\lambda_1, \cdots, \lambda_n)$ is the diagonal matrix of the eigenvalues of the covariance matrix ordered from highest to lowest:

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}$$

Finally, the principal components are the row vectors of $\mathbf{U^T}$. $\mathbf{U^T}$ represents the projection weight matrix W and the transformed data matrix S can be obtained from the original data matrix X by:

$$\mathbf{S} = \mathbf{WX}$$

If we choose not to use eigenvectors that correspond to lower eigenvalues so that W has fewer rows, then each s will have lower dimensionality regarding its corresponding x. Discarding these eigenvectors can be thought of as discarding noise from the data, since these eigenvectors represent highly correlated, and thus uninformative variables.

2.3.2 Support Vector Machines

SVMs are a useful technique for data classification. Although SVM is considered easier to use than Artificial Neuronal Networks (ANN), users not familiar with it often get unsatisfactory results at first. Furthermore, there are two main advantages of SVM over ANN. First, most of the modalities of ANN can suffer from multiple local minima while the solution supplied by SVM is global and unique. Second, the computational complexity of

SVM does not depend on the input data dimensionality, unlike ANN. Although readers do not need to understand the underlying theory behind SVM, the basics necessary for explaining the used SVM package are given. In this case, the libsvm library (free software available in Internet [CL11]) has been employed and the brief following explanation is adapted from this article.

A classification task usually implies separating data into training and testing sets. Each instance in the training is added to one class group and contains several "attributes" (i.e. the features or observed variables). The goal of SVM is to produce a model (based on the training data) which predicts the class label of the test data, given only the test data attributes.

Given a training set of instance-label pairs $(x_i, y_i), i = 1, ..., l$ where $x_i \in R^n$ and $y_i \in \{1, -1\}^l$, the SVM require the solution of the following optimization problem ([BGV92] and [CV95]):

$$\begin{array}{l} \min\limits_{\mathbf{w},b,\xi} \ \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=0}^{l}\xi_i \\ \text{subject to } y_i(\mathbf{w}^T\phi(x_i) + b) \geq 1 - \xi_i, \\ \xi_i \geq 0. \end{array}$$

Here, training vectors x_i are mapped into a higher (maybe infinite) dimensional space by the function ϕ. SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. $C > 0$ is the penalty parameter of the error term. Furthermore, $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$ is called the kernel function. Though new kernels are being proposed by researchers, the following four basic kernels are easily found in literature:

- linear: $K(x_i, x_j) = x_i^T x_j$.
- polynomial: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$.
- radial basis function (RBF): $K(x_i, x_j) = exp(-\gamma\|x_i - x_j\|^2), \gamma > 0$.
- sigmoid: $K(x_i, x_j) = \tanh(x_i^T x_j + r)$.

2.4 Fuzzy Logic

The main reasoning system used in the different works of this thesis is FL. Therefore, the basis of FL will now be presented, namely the concepts that are mostly used. These concepts are based on the explanations found in [Ful10] and [Kae] but readers interested in other questions concerning FL may consult the following references [Zad75], [Zad99], [DD96] and [YF94].

FL was conceived by Lofti Zadeh as a way of processing data by allowing partial set membership rather than crisp set membership or non-membership. According to him, people do not require precise, numerical information input, and yet they are capable of highly adaptive control. He believed that feedback controllers could be programmed in a way that they would be able to accept noisy and/or imprecise input and this way they could be more effective and even easier to implement.

It is possible to see FL as problem-solving control system methodology that can be implemented in a wide range of different systems. It can be implemented in hardware, software or both. FL provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information. A FL approach to control problems mimics how a person would make decisions, only much faster.

Among the benefits of using FL there are some features that we would like to highlight:

- It is considered to be robust as it does not require precise and noise-free inputs. Even in case of an input failure it may continue to work and its output is normally smooth despite the several possible inputs.

- It is very modular and easily adaptable. As a matter of fact, its rules can be easily changed and tuned to drastically change the system performance. New sensors can be easily incorporated by setting up new rules and/or adapting he existing ones.

- It allows the use of a wide range of different types of sensors as it is not limited to a few feedback inputs and one or two control outputs, nor is it necessary to measure or compute rate-of-change parameters in order for it to be implemented. It is then possible to use inexpensive and even imprecise sensors while keeping the overall system cost and complexity low.

- Any reasonable number of inputs can be processed (1-8 or more) and numerous outputs (1-4 or more) generated. Nevertheless, the complexity of the rule-base may increase when using many inputs so it is advisable to distribute different sub tasks to different controllers (using, for instance, a Hierarchical Fuzzy System(s) (HFS) approach).

- It is also possible to model nonlinear system thus making it possible to model control systems that would normally be deemed unfeasible for automation.

Professor Lotfi Zadeh proposed the concept of linguistic or "fuzzy" variables. We can see them as linguistic objects or words, rather than numbers. The sensor input is a noun, e.g. "temperature", "displacement", "velocity", "flow", "pressure", etc. Since error is just the difference, it can be thought of the same way. The fuzzy variables themselves are adjectives that modify the variable (e.g. "large positive" error, "small positive" error, "zero" error, "small negative" error, and "large negative" error). As a minimum, one could simply have "positive", "zero", and "negative" variables for each of the parameters. Additional ranges such as "very large" and "very small" could also be added to extend the responsiveness to exceptional or very nonlinear conditions, but aren't necessary in a basic system.

FL incorporates a simple, rule-based *IF X AND Y THEN Z* approach to a solving control problem rather than attempting to model a system mathematically. The FL model is empirically-based, relying on an operator experience rather than their technical understanding of the system. For instance, rather than dealing with temperature control in terms such as "SP = 500F", "T < 1000F", or "210C < TEMP < 220C", terms like *"IF (process is too cool) AND (process is getting colder) THEN (add heat to the process)"* or *"IF (process is too hot) AND (process is heating rapidly) THEN (cool the process quickly)"* are used. These terms are imprecise and yet very descriptive of what must actually happen. Consider what you do in the shower if the temperature is too cold: you will make the water comfortable very quickly with little trouble. FL is capable of mimicking this type of behavior but at very high rate.

The next logical question is how to apply the rules. This leads into the next concept, the membership function. The membership function is a graphical representation of the magnitude of participation of each input. It associates a weighting with each of the inputs that are processed, defines functional overlap between inputs, and ultimately determines an output response. The rules use the input membership values as weighting factors to determine their influence on the fuzzy output sets of the final output conclusion. There are different membership functions associated with each input and output response.

The logical products for each rule must be combined or inferred before being passed on to the defuzzification process for crisp output generation. Once the functions are inferred, scaled, and combined, they are defuzzified into a crisp output which drives the system by combining the results of the inference process and then computing the "fuzzy centroid" of the area.

The system should be tuned in order to produce the best results. This can be done by changing the rule antecedents or conclusions, changing the centers of the input and/or output membership functions, or adding additional degrees to the input and/or output functions such as “low”, “medium”, and “high” levels of “error”, “error-dot”, and output response. These new levels would generate additional rules and membership functions which would overlap with adjacent functions forming longer “mountain ranges” of functions and responses. The techniques for doing this systematically are a subject unto itself.

After this brief explanation of FL, and as this technique is widely used on this thesis, we will describe some of the previous mentioned concepts in a formal way, namely those which are most used in our work.

2.4.1 Fuzzy sets

According to the definition of [Zad65], let X be a nonempty set. A fuzzy set A in X is characterized by its membership function

$$\mu_{\mathbf{A}} : \mathbf{X} \to [\mathbf{0}, \mathbf{1}]$$

and $\mu_A(x)$ is interpreted as the degree of membership of element x in fuzzy set A for each $x \in X$.

Frequently we will write simply $A(x)$ instead of $\mu_A(x)$.

Let A be a fuzzy subset of X; the support of A, denoted *supp*(A), is the crisp subset of X whose elements all have nonzero membership grades in A.

2.4.2 Fuzzy Numbers

A fuzzy set A of the real line $\mathbb{R}$ is defined by its membership function (denoted also by A)

$$\mathbf{A} : \mathbb{R} \to [\mathbf{0}, \mathbf{1}]$$

If $x \in \mathbb{R}$ then $A(x)$ is interpreted as the degree of membership of x in A.

A fuzzy set in $\mathbb{R}$ is called normal if there exists an $x \in \mathbb{R}$ such that $A(x) = 1$. A fuzzy set in $\mathbb{R}$ is said to be convex if A is unimodal (as a function). A fuzzy number A is a fuzzy set of the real line with a normal, (fuzzy) convex and continuous membership function of bounded support.

Definition: A fuzzy set A is called triangular fuzzy number with peak (or center) a, left width $\alpha > 0$ and right width $\beta > 0$ if its membership function has the following form:

$$A(t) = \begin{cases} 1 - \frac{a-t}{\alpha} & \text{if } a - \alpha \le t \le a \\ 1 - \frac{t-a}{\beta} & \text{if } a \le t \le a + \beta \\ 0 & \text{otherwise} \end{cases}$$

and we use the notation $A = (a, \alpha, \beta)$.

The support of A is $(a-\alpha, b+\beta)$. A triangular fuzzy number with center a may be seen as a fuzzy quantity

"x is close to a" or "x is approximately equal to a":

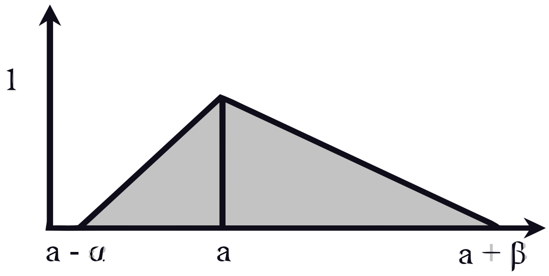


Figure 2.5: A triangular fuzzy number

Definition: A fuzzy set of the real line given by the membership function

$$A(t) = \begin{cases} 1 - \frac{|a-t|}{\alpha} & \text{if } |a - t| \le \alpha, \\ 0 & \text{otherwise} \end{cases}$$

where $\alpha > 0$ will be called a symmetrical triangular fuzzy number with center $a \in \mathbb{R}$ and width 2α and we shall refer to it by the pair (a, α).

Definition: A fuzzy set A is called trapezoidal fuzzy number with tolerance interval $[a, b]$, left width α and right width β if its membership function has the following form:

$$A(t) = \begin{cases} 1 - \frac{a-t}{\alpha} & \text{if } a - \alpha \le t \le a \\ 1 & \text{if } a \le t \le b \\ 1 - \frac{t-b}{\beta} & \text{if } a \le t \le b + \beta \\ 0 & \text{otherwise} \end{cases}$$

and we use the notation $A = (a, b, \alpha, \beta)$

The support of A is $(a - \alpha, b + \beta)$. A trapezoidal fuzzy number may be seen as a fuzzy quantity

"x is approximately in the interval $[a, b]$":

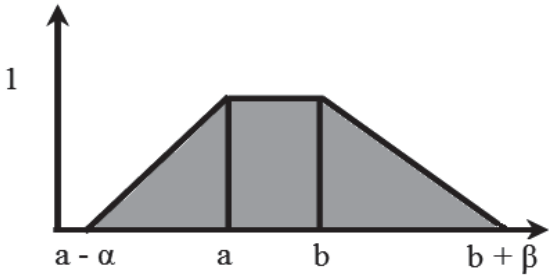


Figure 2.6: Trapezoidal fuzzy number

2.4.3 Operations on Fuzzy Sets

The classical set theoretic operations from ordinary set theory can be extended to fuzzy sets. Let A and B are fuzzy subsets of a crisp set X. The classical - introduced by Zadeh in 1965 - intersection of A and B is defined as:

$(A \cap B)(t) = \min\{A(t), B(t)\} = A(t) \wedge B(t),$

The union of A and B is defined as

$(A \cup B)(t) = \max\{A(t), B(t)\} = A(t) \vee B(t),$

The complement of a fuzzy set A is defined as

$(\neg A)(t) = 1 - A(t)$

for all $t \in X$.

2.4.4 Triangular Norms

Triangular norms were introduced by Schweizer and Sklar to model distances in probabilistic metric spaces. In fuzzy sets theory triangular norms are extensively used to model logical connective *and*.

Definition: (Triangular norm.) A mapping

$\mathbf{T} : [0,1] \times [0,1] \to [0,1]$

is a triangular norm (t-norm for short) iff it is symmetric, associative, non-decreasing in each argument and $\mathbf{T}(a,1) = a$, for all $a \in [0,1]$. In other words, any t-norm $\mathbf{T}$ satisfies the properties:

- Symmetricity: $\mathbf{T}(x,y) = \mathbf{T}(y,x), \forall x,y \in [0,1]$.
- Associativity: $\mathbf{T}(x,\mathbf{T}(y,z)) = \mathbf{T}(\mathbf{T}(x,y),z), \forall x,y,z \in [0,1]$.
- Monotonicity: $\mathbf{T}(x,y) \leq \mathbf{T}(x',y')$ if $x \leq x'$ and $y \leq y'$.
- One identy: $\mathbf{T}(x,1) = x, \forall x \in [0,1]$.

These axioms attempt to capture the basic properties of set intersection. The basic t-norms are:

- minimum: $\min(a,b) = \min\{a,b\}$,
- Lukasiewicz: $\mathbf{T}_L(a,b) = \max\{a+b-1,0\}$
- product: $\mathbf{T}_P(a,b) = ab$

2.4.5 Triangular conorms

Triangular conorms are extensively used to model logical connective *or*.

Definition: (Triangular conorm.) A mapping

$\mathbf{S} : [0,1] \times [0,1] \to [0,1]$

is a triangular co-norm (t-conorm) if it is symmetric, associative, non-decreasing in each argument and $\mathbf{S}(a,0) = a$, for all $a \in [0,1]$. In other words, any t-conorm $\mathbf{S}$ satisfies the properties:

- Symmetricity: $\mathbf{S}(x,y) = \mathbf{S}(y,x), \forall x,y \in [0,1]$.
- Associativity: $\mathbf{S}(x,\mathbf{S}(y,z)) = \mathbf{S}(\mathbf{S}(x,y),z), \forall x,y,z \in [0,1]$.
- Monotonicity: $\mathbf{S}(x,y) \leq \mathbf{S}(x',y')$ if $x \leq x'$ and $y \leq y'$
- Zero Identy: $\mathbf{S}(x,0) = x, \forall x \in [0,1]$

If $\mathbf{T}$ is a t-norm then the equality

$\mathbf{S}(a,b) := 1 - \mathbf{T}(1-a, 1-b),$

defines a t-conorm and we say that $\mathbf{S}$ is derived from $\mathbf{T}$.
The basic t-conorms are:

- maximum: $\max(a,b) = \max\{a,b\}$,
- Lukasiewicz: $\mathbf{S}_L(a,b) = \min\{a+b, 1\}$
- product: $\mathbf{S}_P(a,b) = a + b - ab$

2.4.6 Material Implication

Let $p ='\ x$ is in A' and $q ='$ y is in B' are crisp propositions, where A and B are crisp sets for the moment.

The full interpretation of the material implication $p \to q$ is that:

the degree of truth of $p \to q$ quantifies to what extend q is at least as true as p, i.e.

$$\tau(p \to q) = \begin{cases} 1 & \text{if } \tau(p) \leq \tau(q) \\ 0 & \text{otherwise} \end{cases}$$

2.4.7 Fuzzy Implications

Consider the implication statement

"if pressure is high then volume is small"

The membership function of the fuzzy set A, *big pressure*, can be interpreted as

- 1 is in the fuzzy set *big pressure* with grade of membership 0
- 2 is in the fuzzy set *big pressure* with grade of membership 0.25
- 4 is in the fuzzy set *big pressure* with grade of membership 0.75
- x is in the fuzzy set *big pressure* with grade of membership 1, $x \geq 5$

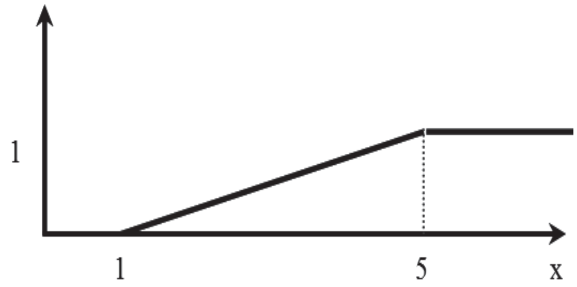


Figure 2.7: Membership function for "big pressure"

$$A(u) = \begin{cases} 1 & \text{if } u \geq 5 \\ 1 - \frac{5-u}{4} & \text{if } 1 \leq u \leq 5 \\ 0 & \text{otherwise} \end{cases}$$

The membership function of the fuzzy set B, *small volume*, can be interpreted as:

- 5 is in the fuzzy set *small volume* with grade of membership 0
- 4 is in the fuzzy set *small volume* with grade of membership 0.25
- 2 is in the fuzzy set *small volume* with grade of membership 0.75
- x is in the fuzzy set *small volume* with grade of membership 1, $x \leq 1$

$$B(v) = \begin{cases} 1 & \text{if } v \leq 1 \\ 1 - \frac{v-1}{4} & \text{if } 1 \leq v \leq 5 \\ 0 & \text{otherwise} \end{cases}$$

If p is a proposition of the form "x is A" where A is a fuzzy set, for example, *big pressure* and q is a proposition of the form "y is B" for example, *small volume* then we define the implication $p \to q$ as

$$A(x) \to B(y)$$

For example,

x is big pressure $\to$ y is small volume $\equiv A(x) \to B(y)$

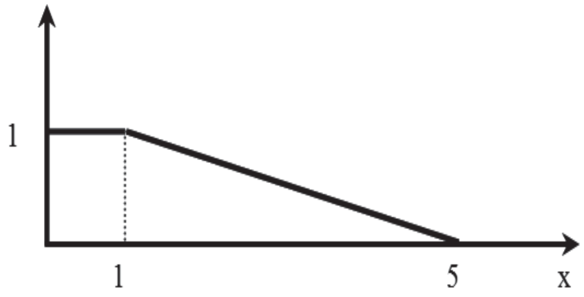


Figure 2.8: Membership function for “small volume”

Remembering the full interpretation of the material implication

$$p \to q = \begin{cases} 1 & \text{if } \tau(p) \le \tau(q) \\ 0 & \text{otherwise} \end{cases}$$

We can use the definition

$$A(x) \to B(y) = \begin{cases} 1 & \text{if } A(x) \le B(x) \\ 0 & \text{otherwise} \end{cases}$$

4 is big pressure → 1 is small volume $= A(4) \to B(1) = 1$.

In many practical applications they use Mamdani's minimum operator to model causal relationship between fuzzy variables.

$$A(u) \to B(v) = \min\{A(u), B(v)\}$$

For example,

4 is big pressure → 1 is small volume $= \min\{A(4), B(1)\} = 0.75$

It is easy to see this is not a correct extension of material implications, because $0 \to 0$ yields zero. However, in knowledge based systems, we are usually not interested in rules, where the antecedent part is false.

2.4.8 The Theory of Approximate Reasoning

In 1979 Zadeh introduced the theory of approximate reasoning. This theory provides a powerful framework for reasoning in the face of imprecise and uncertain information. Central to this theory is the representation of propositions as statements assigning fuzzy sets as values to variables.

Suppose we have two interactive variables $x \in X$ and $y \in Y$ and the causal relationship between x and y is completely known. Namely, we know that y is a function of x

$$y = f(x)$$

Then we can make inferences easily

$$\frac{\begin{array}{c}\text{premise } y = f(x) \\ \text{fact } x = x'\end{array}}{\text{consequence } y = f(x')}$$

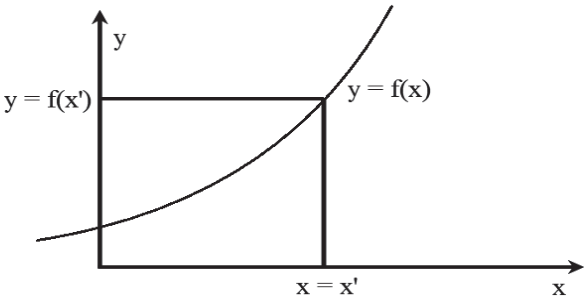


Figure 2.9: Simple crisp inference

This inference rule says that if we have $y = f(x), \forall x \in X$ and we observe that $x = x'$ then y takes the value of $f(x')$.

More often than not we do not know the complete causal link f between x and y, only we now the values of $f(x)$ for some particular values of x,

$$\begin{array}{c}\Re_1: \text{ if } x = x_1 \text{ then } y = y_1 \\ \Re_2: \text{ if } x = x_2 \text{ then } y = y_2 \\ \dots \\ \Re_n: \text{ if } x = x_n \text{ then } y = y_n\end{array}$$

Suppose that we are given an $x' \in X$ and want to find an $y' \in Y$ which corresponds to x' under the rule-base $\{\Re_1, \dots, \Re_n\}$,

$$\begin{array}{rl}
\Re_1: & \text{if } x = x_1 \text{ then } y = y_1 \\
\Re_2: & \text{if } x = x_2 \text{ then } y = y_2 \\
 & \cdots \\
\Re_n: & \text{if } x = x_n \text{ then } y = y_n \\
\text{fact:} & \qquad\qquad x = x' \\
\hline
\text{consequence:} & \qquad\qquad y = y'
\end{array}$$

This problem is frequently quoted as interpolation.

Let x and y be linguistic variables, e.g. "x is high" and "y is small". The basic problem of approximate reasoning is to find the membership function of the consequence C from the rule-base $\{\Re_1, \dots, \Re_n\}$ and the fact A,

$$\begin{array}{rl}
\Re_1: & \text{if } x \text{ is } A_1 \text{ then } y \text{ is } C_1 \\
\Re_2: & \text{if } x \text{ is } A_2 \text{ then } y \text{ is } C_2 \\
 & \cdots \\
\Re_n: & \text{if } x \text{ is } A_n \text{ then } y \text{ is } C_n \\
\text{fact:} & \qquad\qquad x \text{ is } A \\
\hline
\text{consequence:} & \qquad\qquad y = C
\end{array}$$

In 1979 Zadeh introduces a number of translation rules which allow us to represent some common linguistic statements in terms of propositions in our language. In the following we describe some of these translation rules.

Definition: Entailment rule:

$$\begin{array}{rl}
x \text{ is } A: & \text{Mary is very young} \\
A \subset B: & \text{very young} \subset \text{young} \\
\hline
x \text{ is } B: & \qquad \text{Mary is young}
\end{array}$$

Definition: Conjunction rule:

$$\begin{array}{c}
x \text{ is } A \\
x \text{ is } B \\
\hline
x \text{ is } A \cap B
\end{array}$$

Example

$$\begin{array}{c}
\text{pressure is not very high} \\
\text{pressure is not very low} \\
\hline
\text{pressure is not very high and not very low}
\end{array}$$

Definition: Disjunction rule:

$$\frac{\begin{array}{c} x \text{ is } A \\ x \text{ is } B \end{array}}{x \text{ is } A \cup B}$$

Example

$$\frac{\begin{array}{c} \text{pressure is not very high} \\ \text{pressure is not very low} \end{array}}{\text{pressure is not very high or not very low}}$$

Definition: Projection rule:

$$\frac{(x, y) \text{ have relation } R}{x \text{ is } \Pi_X(R)}$$

$$\frac{(x, y) \text{ have relation } R}{y \text{ is } \Pi_Y(R)}$$

Example

$$\frac{(x, y) \text{ is close to } (3, 2)}{x \text{ is close to } 3}$$

$$\frac{(x, y) \text{ is close to } (3, 2)}{y \text{ is close to } 2}$$

Definition: Negation rule:

$$\frac{\text{not (x is A)}}{x \text{ is } \neg A}$$

Example

$$\frac{\text{not (x is high)}}{x \text{ is not high}}$$

In fuzzy logic and approximate reasoning, the most important fuzzy implication inference rule is the Generalized Modus Ponens (GMP). The classical Modus Ponens inference rule says:

$$\begin{array}{ll} \text{premise} & \text{if } p \text{ then } q \\ \text{fact} & p \\ \hline \text{consequence} & q \end{array}$$

This inference rule can be interpreted as: If p is true and $p \to q$ is true then q is true.

The fuzzy implication inference is based on the compositional rule of inference for approximate reasoning suggested by Zadeh in 1973.

Definition: (compositional rule of inference)

$$\begin{array}{lr} \text{premise} & \text{if } x \text{ is } A \text{ then } y \text{ is } B \\ \text{fact} & x \text{ is } A' \\ \hline \text{consequence} & y \text{ is } B' \end{array}$$

where the consequence B' is determined as a composition of the fact and the fuzzy implication operator

$$B' = A' \circ (A \to B)$$

that is,

$$B'(v) = \sup_{u \in U} \min\{A'(u), (A \to B)(u, v)\}, v \in V.$$

The consequence B' is nothing else but the shadow of $A \to B$ on A'.

The Generalized Modus Ponens, which reduces to classical modus ponens when $A' = A$ and $B' = B$, is closely related to the forward data-driven inference which is particularly useful in the Fuzzy Logic Control.

In many practical cases instead of sup-min composition we use sup-**T** composition, where **T** is a t-norm.

Definition: (sup-**T** compositional rule of inference)

$$\begin{array}{lr} \text{premise} & \text{if } x \text{ is } A \text{ then } y \text{ is } B \\ \text{fact} & x \text{ is } A' \\ \hline \text{consequence} & y \text{ is } B' \end{array}$$

where the consequence B' is determined as a composition of the fact and the fuzzy implication operator

$$B' = A' \circ (A \to B)$$

that is,

$$B'(v) = \sup\{\mathbf{T}(A'(u), (A \to B)(u, v)) | u \in U\}, v \in V$$

It is clear that **T** can not be chosen independently of the implication operator.

The classical Modus Tollens inference rule says: If $p \to q$ is true and q is false then p is false. The Generalized Modus Tollens,

$$\begin{array}{lr} \text{premise} & \text{if } x \text{ is } A \text{ then } y \text{ is } B \\ \text{fact} & y \text{ is } B' \\ \hline \text{consequence} & x \text{ is } A' \end{array}$$

which reduces to "Modus Tollens" when $B = \neg B$ and $A' = \neg A$, is closely related to the backward goal-driven inference which is commonly used in expert systems, especially in the realm of medical diagnosis.

Suppose that A, B and A' are fuzzy numbers. The Generalized Modus Ponens should satisfy the basic property:

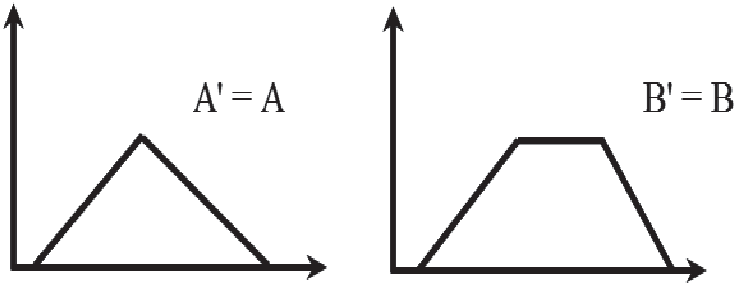


Figure 2.10: Basic property

$$\begin{array}{lr} \text{premise} & \text{if } x \text{ is } A \text{ then } y \text{ is } B \\ \text{fact} & x \text{ is } A \\ \hline \text{consequence} & y \text{ is } B \end{array}$$

Example

$$\begin{array}{lr} \text{if pressure is big then volume is small} \\ \quad \text{pressure is big} \\ \hline \qquad\qquad\qquad\qquad\qquad \text{volume is small} \end{array}$$

2.4.9 Simplified Fuzzy Reasoning Schemes

Suppose that we have the following rule base

$$
\begin{array}{ll}
\Re_1: & \text{if } x \text{ is } A_1 \text{ then } y \text{ is } z_1 \\
 & \qquad\text{also} \\
\Re_2: & \text{if } x \text{ is } A_2 \text{ then } y \text{ is } z_2 \\
 & \qquad\dots \\
\Re_n: & \text{if } x \text{ is } A_n \text{ then } y \text{ is } z_n \\
\text{fact:} & \hfill x \text{ is } x_0 \\
\hline
\text{action:} & \hfill y \text{ is } z_0
\end{array}
$$

where $(A_1, \dots, A_n)$ are fuzzy sets.

Suppose further that our data base consists of a single fact x_0. The problem is to derive z_0 from the initial content of the data base, x_0, and from the fuzzy rule base $\Re = \{\Re_1, \dots, \Re_n\}$.

$$
\begin{array}{ll}
\Re_1: & \text{if salary is small then loan is } z_1 \\
 & \qquad\text{also} \\
\Re_2: & \text{if salary is big then loan is } z_2 \\
\text{fact:} & \hfill \text{salary is } x_0 \\
\hline
\text{action:} & \hfill \text{loan is } z_0
\end{array}
$$

A deterministic rule base can be formed as follows

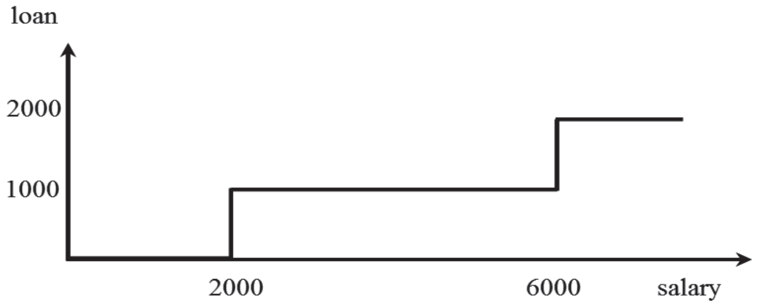


Figure 2.11: Discrete causal link between “salary” and “loan”.

$$
\begin{array}{lll}
\Re_1: & \text{if } 2000 \le s \le 6000 & \text{then loan is max } 1000 \\
\Re_2: & \text{if } s \ge 6000 & \text{then loan is max } 2000 \\
\Re_3: & \text{if } s \le 2000 & \text{then} \qquad \text{no loan at all}
\end{array}
$$

The data base contains the actual salary, and then one of the rules is applied to obtain the maximal loan can be obtained by the applicant.

In fuzzy logic everything is a matter of degree.

If x is the amount of the salary then x belongs to fuzzy set

- A_1 = small with degree of membership $0 \leq A_1(x) \leq 1$
- A_2 = big with degree of membership $0 \leq A_2(x) \leq 1$

In fuzzy rule-based systems each rule fires.

The degree of match of the input to a rule (wich is the firing strength) is the membership degree of the input in the fuzzy set characterizing the antecedent part of the rule.

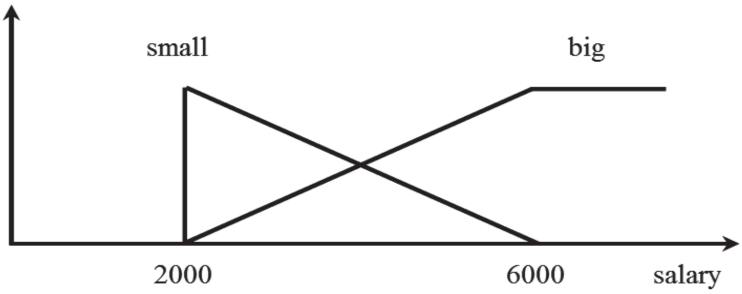


Figure 2.12: Membership functions for “small” and “big”.

The overall system output is the weighted average of the individual rule outputs, where the weight of a rule is its firing strength with respect to the input.

To illustrate this principle we consider a very simple example mentioned above

$$\begin{array}{ll} \Re_1: & \text{if salary is small then loan is } z_1 \\ & \text{also} \\ \Re_2: & \text{if salary is big then loan is } z_2 \\ \text{fact:} & \text{salary is } x_0 \\ \hline \text{action:} & \text{loan is } z_0 \end{array}$$

Then our reasoning system is the following

- input to the system is x_0
- the firing level of the first rule is $\alpha_1 = A_1(x_0)$
- the firing level of the first rule is $\alpha_2 = A_2(x_0)$

- the overall system output is computed as the weighted average of the individual rule outputs
  $z_0 = \frac{\alpha_1 z_1 + \alpha_2 z_2}{\alpha_1 + \alpha_2}$ that is $z_0 = \frac{A_1(x_0) z_1 + A_2(x_0) z_2}{A_1(x_0) + A_2(x_0)}$

$$A_1(x_0) = \begin{cases} 1 - (x_0 - 2000)/4000 & \text{if } 2000 \le x_0 \le 6000 \\ 0 & \text{otherwise} \end{cases}$$

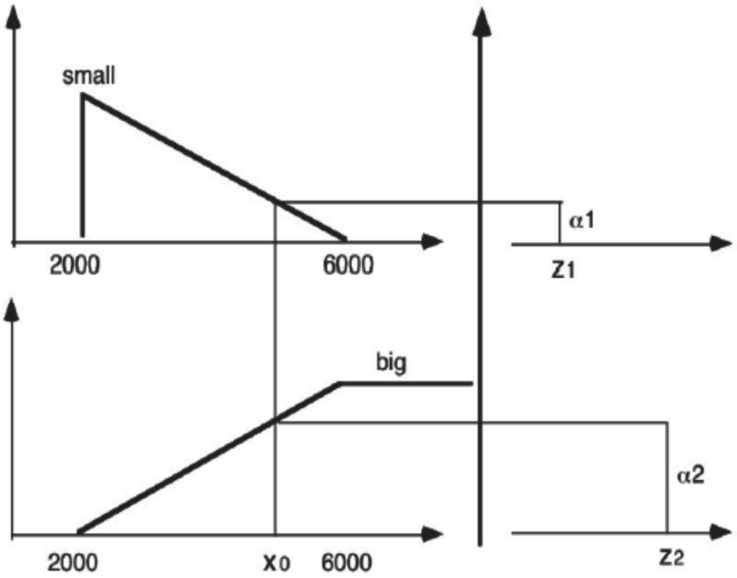


Figure 2.13: Example of simplified fuzzy reasoning.

$$A_2(x_0) = \begin{cases} 1 & \text{if } x_0 \ge 6000 \\ 1 - (6000 - x_0)/4000 & \text{if } 2000 \le x_0 \le 6000 \\ 0 & \text{otherwise} \end{cases}$$

It is easy to see that the relationship

$A_1(x_0) + A_2(x_0) = 1$

holds for all $x_0 \ge 2000$. It means that our system output can be written in the form.

$z_0 = \alpha_1 z_1 + \alpha_2 z_2 = A_1(x_0) z_1 + A_2(x_0) z_2$

that is,

$z_0 = (1 - \frac{x_0 - 2000}{4000})z_1 + (1 - \frac{6000 - x_0}{4000})z_2$

if $2000 \le x_0 \le 6000$. And $z_0 = 1$ if $x_0 \ge 6000$. And $z_0 = 0$ if $x_0 \le 2000$

The (linear) input/oputput relationship is illustrated in the following figure.

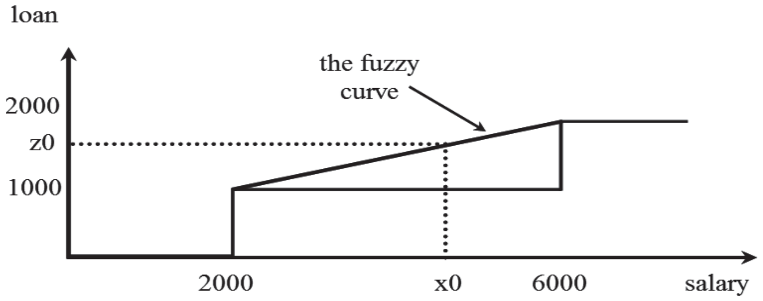


Figure 2.14: Input/output function derived from fuzzy rules.

2.4.10 Fuzzy Reasoning Schemes

$$\begin{array}{rl}
\Re_1: & \text{if } x \text{ is } A_1 \text{ and } y \text{ is } B_1 \text{ then } z \text{ is } C_1 \\
\Re_2: & \text{if } x \text{ is } A_2 \text{ and } y \text{ is } B_2 \text{ then } z \text{ is } C_2 \\
& \cdots \\
\Re_n: & \text{if } x \text{ is } A_n \text{ and } y \text{ is } B_n \text{ then } z \text{ is } C_n \\
& x \text{ is } \bar{x}_0 \text{ and } y \text{ is } \bar{y}_0 \\
\hline
& \hfill z \text{ is } C
\end{array}$$

The i-th fuzzy rule from this rule-base

$\Re_i:$ if x is A_i and y is B_i then z is C_i

is implemented by a fuzzy relation R_i and is defined as

$R_i(u, v, \omega) = (A_i \times B_i \to C_i)(u, \omega) = [A_i(u) \wedge B_i(v)] \to C_i(\omega)$

for $i = 1, \dots, n$.

Find C from the input x_0 and from the rule base $\Re = \{\Re_1, \dots, \Re_n\}$.

Interpretation of

- logical connective "and"
- sentence connective "also"

- implication operator "then"
- compositional operator "$\circ$"

We first compose $\bar{x}_0 \times \bar{y}_0$ with each R_i producing intermediate result

$$C'_i = \bar{x}_0 \times \bar{y}_0 \circ R_i$$

for $i = 1, \ldots, n$. Here C'_i is called the output of the i-th rule

$$C'_i(\omega) = [A_i(x_0) \wedge B_i(y_0)] \rightarrow C_i(\omega)$$

for each ω.

Then combine the C'_i component wise into C' by some aggregation operator:

$$C = \bigcup_{i=1}^{n} C'_i = \bar{x}_0 \times \bar{y}_0 \circ R_1 \cup \cdots \cup \bar{x}_0 \times \bar{y}_0 \circ R_n$$

$$C(\omega) = A_1(x_0) \times B_1(y_0) \rightarrow C_1(\omega) \vee \cdots \vee A_n(x_0) \times B_n(y_0) \rightarrow C_n(\omega)$$

- input to the system is (x_0, y_0)
- fuzzified input is $(\bar{x}_0, \bar{y}_0)$
- firing strength of the i-th rule is $A_i(x_0) \wedge B_i(y_0)$
- the i-th individual rule output is $C'_i(\omega) := [A_i(x_0) \wedge B_i(y_0)] \rightarrow C_i(\omega)$
- overall system output is $C = C'_1 \cup \cdots \cup C'_n$.

overall system output = union of the individual rule outputs.

Among some of the most well-known inference mechanism systems in fuzzy rule-based systems there are the ones of "Mamdani", "Tsukamoto", "Sugeno" and "Larsen".

In the following paragraph there is a brief description about the "Mamdani" inference system. For the other inference systems we advise readers to consult the mentioned bibliography.

For simplicity we assume that we have two fuzzy IF-THEN rules of the form

$$\begin{array}{lr}\Re_1: \text{ if } x \text{ is } A_1 \text{ and } y \text{ is } B_1 \text{ then } z \text{ is } C_1 \\ \multicolumn{2}{c}{\text{also}} \\ \Re_2: \text{ if } x \text{ is } A_2 \text{ and } y \text{ is } B_2 \text{ then } z \text{ is } C_2 \\ \text{fact}: \quad & x \text{ is } \bar{x}_0 \text{ and } y \text{ is } \bar{y}_0 \\ \hline \text{consequence}: & z \text{ is } C \end{array}$$

2.4.11 Mamdani

The fuzzy implication is modelled by Mamdani's minimum operator and the sentence connective *also* is interpreted as *oring* the propositions and defined by max operator.

The firing levels of the rules, denoted by $\alpha_i, i = 1, 2$, are computed by

$$\alpha_1 = A_1(x_0) \wedge B_1(y_0),$$
$$\alpha_2 = A_2(x_0) \wedge B_2(y_0)$$

The individual rule outputs are obtained by

$$C_1'(\omega) = (\alpha_1 \wedge C_1(\omega)),$$
$$C_2'(\omega) = (\alpha_2 \wedge C_2(\omega))$$

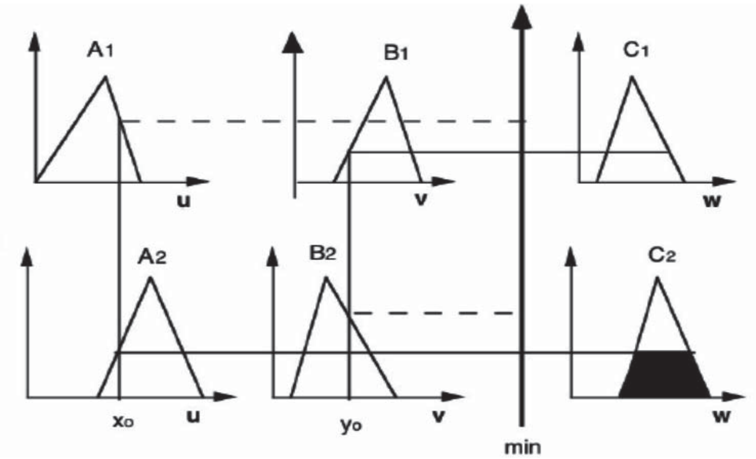


Figure 2.15: Inference with Mamdani's minimum operation rule.

Then the overall system output is computed by *oring* the individual rule outputs

$$C(\omega) = C_1'(\omega) \vee C_2'(\omega) = (\alpha_1 \wedge C_1(\omega)) \vee (\alpha_2 \wedge C_2(\omega))$$

Finally, to obtain a deterministic control action, defuzzification strategy is employed.

Chapter 3

People Detection and Tracking

This Chapter shows a Stereo Vision (SV) system capable of detecting and tracking several people. This task is essential for the posterior detection of the interest and of the human response. As already mentioned in Section 1.3, many research works have presented different approaches to solve different problems during the process. Some of these problems are: the correct detection of human beings, the correct tracking with a minimum loss of the tracked people, the use of human alike sensors and techniques to perform the detection and tracking (enhancing the sense of a more "natural" or "human similar" interaction).

In our case, the detection of people is based on the integration of colour and distance information. On the one hand, independent objects (*blobs*) are detected on the disparity image. They will be marked as possible people. On the other hand a face detector is also applied. These items are merged in order to detect the visible people. To perform the tracking, hair color, clothes and past history information of the located people, are used. This way, people can be identified even though some occlusions occur. Almost the totality of the techniques used to solve the problem of people detection and tracking, namely those described in Section 1.3, are based on probabilistic approaches. In our case we firstly present a probabilistic approach to solve the problem of people tracking. We integrate not only colour and distance information, using a probabilistic approach, but we also use an analogous to the human vision system (stereo camera) which is placed at a similar height comparing to the human vision system (eyes).

We continue by presenting a people detection and tracking algorithm which is based on a "possibilistic" (using Fuzzy Logic (FL)) approach. The reason for proposing a "possibilistic" approach, is because FL allows adding more information based on expert knowledge, when evaluating the particles, without being confined to the probabilistic models. Although stereo and colour information are also used in this work as sources of information, they are supplied to several hierarchically sorted Fuzzy System(s) (FS), also called Hierarchical Fuzzy System. This is done by generating different particles in the image and then, using a FL approach, computing their possibility of being the face central pixel of some previously detected person. In complex applications, containing a large set of variables, it is not appropriate to define the system with a flat set of rules. Among other problems, the number of rules increases exponentially with the number of variables. Thus, FS should be organised according to the type of information they cope with in a hierarchical structure which has the advantage of helping reducing the complexity of such systems ([Tor02]). That is a non negligible advantage that was taken into consideration when choosing the architecture of the employed FS.

In this Chapter, the first detection probabilistic approach is described in Section 3.1 and then we describe the "possibilistic" approach in Section 3.2.

3.1 Probabilistic Human Tracking Approach

A probabilistic approach for tracking people, based on Particle Filter (PF) and intended for mobile robots, will be firstly presented.

3.1.1 Proposed Method

This section explains our probabilistic person tracking method. It is based on the use of the 3D body model shown in Fig. 3.1(a). It is a model comprised by two planar ellipses and the information projected inside them: one fitting the head region of the person (E^h), and another one fitting their torso (E^t). In an initial phase, the model must be appropriately placed to fit the person head and torso (e.g. using a face detector [YKA02]). Then, two color models (one for each ellipse) are stored to be employed in order to track the person. The HSV color space has been employed in this work because it is relatively invariable to illumination changes.

Our tracking approach employs the Condensation algorithm where particles represent positions and velocities of the 3D model. The 3D position of the

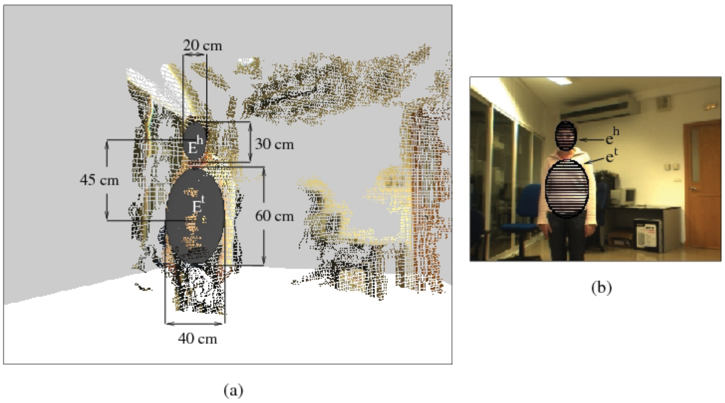


Figure 3.1: (a) Anatomical measures used for the different human body sections, represented over real stereo information corresponding to a scene with a person. (b) Projection of the human model on the reference camera image (shown a person)

model is given by the central position of its upper ellipse $E^h_c = (X, Y, Z)$ that corresponds to the person head being tracked. Given a 3D position for the model, it is possible to determine its projection on the reference camera image. Figure 3.1(b) shows the projection of the 3D model shown in Fig. 3.1(a). Particles weights are calculated by first projecting the 3D model on the reference camera image and then examining the inner pixels of the projected ellipses. If a particle is near the true person location, then the inner pixels of the model projection must have a color distribution similar to the target color models and be at the distance indicated by the particle. Besides, the gradient around the upper ellipse should indicate the presence of an elliptical object (person head). Nevertheless, these assumptions are very strict and several contingencies must be taken into account. First, as previously mentioned, the disparity calculation is subject to errors. Second, in some cases it might be impossible to determine the disparity of the target region because of occlusions or absence of texture. As follows, a detailed explanation about how color and depth information have been combined in this approach in order to deal with these problems, is given.

Initial phase and model projection

The 3D model employed is comprised by two ellipses whose sizes have been selected according to standard people sizes. The ellipses axis lengths are shown in Fig. 3.1(a). Let be E^h_w and E^h_h the horizontal and vertical lengths of the axis of E^h. As it can be seen, the axis of E^t are twice longer than the axis of E^h.

In the initial phase, the model must be appropriately placed to fit the head and torso of the person in order to create the two target color models. They are employed in the tracking phase in order to look for similar colored regions in the subsequent images. The first target color model, named $\hat{q}_{E^t}$, corresponds to the torso of the person being tracked and stores information about the person clothes. The second color model, $\hat{q}_{E^h}$, corresponds to the person head region.

The two ellipsoidal surfaces of the model (E^h and E^t) project in two ellipses on the reference camera image (let us denote them by e^h and e^t). The centre of e^h (let us denote it $e^h_c = (e^h_x, e^h_y)$) is the projection of $E^h_c = (X, Y, Z)$, that can be calculated using projective geometry as:

$$e^h_x = \frac{Xf}{Z};\ e^h_y = -\frac{Yf}{Z}. \tag{3.1}$$

The sizes of the horizontal and vertical axis of e^h, let us denote them e^h_w and e^h_h, can be calculated using projective geometry as:

$$e^h_w = \frac{ZE^h_w}{f};\ e^h_h = \frac{ZE^h_h}{f}. \tag{3.2}$$

The projection of E^t can be calculated using the same procedure, obtaining e^t. The color models of the torso and head projected ellipses (let us denote them by $\hat{q}^t_E$ and $\hat{q}^h_E$ respectively) are stored in order to look for the person in the tracking phase.

Tracking phase

Let a particle $s_i(t) = [X_i(t), Y_i(t), Z_i(t), \dot{X}(t), \dot{Y}(t), \dot{Z}(t)]$ represents the position and speed of the person being tracked. The sample set is propagated using a dynamic model

$$s(t) = As(t-1) + w(t-1), \tag{3.3}$$

where A indicates the deterministic component of the model and $w(t-1)$ is a multivariate Gaussian random variable. We have opted for a first order model where A describes the target moving at constant velocity ($\dot{X}(t)$,$\dot{Y}(t)$,$\dot{Z}(t)$).

As previously indicated, for each particle $s_i(t)$, it is calculated its projection on the reference camera image. Each particle projects as two ellipses $e^h_i(t)$ and $e^t_i(t)$. Our approach consists in examining color and depth of the projected ellipses and the gradient information around the upper one. For the sake of clarity, it is explained first how color and depth information are modelled for each projected ellipse, and then it is explained how gradient information is examined for the upper one.

For approximation both color and depth information are considered as "normal-behaved" because the distribution of the information is expected to be more similar to its neighbourhood than to further information.

Colour information is managed by defining the variable $d_i^h(t) \sim N(0, \sigma_c)$ that is the Bhattacharyya distance:

$$d_i^h(t) = \sqrt{1 - \rho(\hat{q}_E^h, \hat{q}_{e,i}^h(t))}. \tag{3.4}$$

It provides values near 0 when two color models are similar and tends to 1 as they differ. In Eq. 3.4, $\hat{q}_{e,i}^h(t)$ is the color model of $e_i^h(t)$ and $\hat{q}_E^h$ is the target color model of the person head.

Depth information is managed by the variable $\mu_{z,i}^h(t) \sim N(Z_i(t), \sigma_z)$ that is defined as:

$$\mu_{z,i}^h(t) = K \sum_{j=1}^{n} w \left(\frac{||e_{c,i}^h(t) - p_{j,i}^h(t)||}{a} \right) I_z(p_{j,i}^h(t)). \tag{3.5}$$

where I_z represents the *distance image* obtained from the disparity map, each pixel $I_z(p)$ represents the Z component of the point p, w is the weighting function defined as: $w(r) = \begin{cases} 1 - r^2 & if\ r < 1 \\ 0 & otherwise \end{cases}$, a is the distance from the farthest point of the ellipse to its centre $e_{c,i}^h(t)$, K is a normalisation constant calculated by imposing the condition that $\sum_{j=1}^{n} w \left(\frac{||e_{c,i}^h(t) - p_{j,i}^h(t)||}{a} \right) = 1$ and $\{p_{j,i}^h(t)\}_{j=1 \ldots n_i(t)}$ are the inner pixels of $e_i^h(t)$. The variable $\mu_{z,i}^h(t)$ represents the average distance of the pixels enclosed in $e_i^h(t)$, assigning more relevance to central pixels (using w). Assigning more relevance to central pixels helps to reduce the influence of occluding objects in the target boundaries.

It must be reminded that I_z might contain undefined values (unmatched points) so that Eq. 3.5 is only applied for these pixels $p_{j,i}^h(t)$ whose distance is known. Thus, the value provided by $\mu_{z,i}^h(t)$ is affected by uncertainty since there might be unmatched points that if detected might alter its value. The intention is to manage the possible absence of depth information into the model in order to do it more robust. The greater the amount of disparity found, the higher the degree of confidence assigned to depth information is. The problem is then to define a probability distribution function that merges the original distribution taking into account the degree of confidence in $\mu^h_{z,i}(t)$. Our proposal consists in calculating a confidence measure that is included in the standard deviation of the probability distribution function of $\mu^h_{z,i}(t)$. The idea is to modify the shape of the normal distribution so that when the confidence in depth information is high, the new distribution is exactly like the original one. However, as the confidence on depth information decreases, the standard deviation of the probability distribution function is increased making the distribution more similar to an uniform one.

Let us denote as $\lambda^h(t)$ the confidence measure that indicates the proportion of valid points detected in the inner pixels of all the upper projected ellipses $(e^h_i(t)_{i=1..N})$ respect to the total points analysed:

$$\lambda^h(t) = \frac{\sum_{i=1}^{N} \sum_{j=1}^{n_i} \delta(p^h_{j,i}(t))}{\sum_{i=1}^{N} n^i(t)}. \tag{3.6}$$

In Eq. 3.6, N is the number of particles and δ is a function that only has two values: it is 0 when the pixel $p^h_{j,i}(t)$ has an undefined distance value and 1 in the opposite case. Thus, the value $\lambda^h(t)$ is in the range $[0, 1]$, where 1 means that for each particle all the pixels in all the projected ellipses have a known distance value, and decreases to 0 as the number of unmatched points increases.

Using the above calculated $\lambda^h(t)$, the probability distribution of depth information is redefined for $\lambda^h(t) \neq 0$ as:

$$\mu^h_{z,i}(t) \sim N(Z_i(t), \sigma^h_z(t)),$$

where

$$\sigma^h_z(t) = \frac{\sigma_z}{\lambda^h(t)}.$$

The probability distribution goes from a normal with the mass of the probability around $Z_i(t)$ to, little by little, a distribution with all the values with the same probability. So, when $\lambda^h(t) = 0$, $\mu^h_{z,i}(t)$ follows an uniform distribution.

The joint probability distribution function of color and depth for the upper ellipse, when $\lambda^h(t) \neq 0$, is defined as:

$$P_{cd}(e^h_i(t)) = \frac{1}{2\pi\sigma_c\sigma_z(t)} \exp\left(-\frac{1}{2}\left(\frac{d^h_i(t)^2}{\sigma_c^2} + \frac{(\mu^h_{z,i}(t) - Z_i(t))^2}{\sigma_z(t)^2}\right)\right) \tag{3.7}$$

When $\lambda^h(t) = 0$, $\mu^h_{z,i}(t)$ is an uniform and any value is equally probable. In case of total absence of disparity ($\lambda^h(t) = 0$), the hereby presented approach performs as pure color-based tracker.

For the torso ellipse $e_i^t(t)$, a similar reasoning is used in order to define the probability distribution function $P_{cd}(e_i^t(t))$.

Finally, the detection whether the projected ellipse perimeter $e_i^h(t)$ is placed on an ellipsoidal object is achieved by analysing the image gradient. This is a technique employed by several authors in the related literature [BH94, Bir98, BCZ93]. A variant of the Birchfield's method [Bir98] is used that evaluates the gradient direction of the ellipse perimeter. The measure $fitting_i(t)$ is defined as:

$$fitting_i(t) = 1 - \frac{1}{N}\sum_{j=1}^{N} |n_j \cdot g_j|, \tag{3.8}$$

where N is the total number of pixels in the perimeter of the ellipse $e_i^h(t)$, $(\cdot)$ denotes the dot product, g_j is the unit gradient vector of the image at the j-th pixel of the perimeter and u_j is the unit vector normal to the ellipse at pixel j. Assumming $fitting_i(t) \sim N(0, \sigma_g)$, its probability distribution function is defined as:

$$\phi_i(t) = \frac{1}{\sqrt{2\pi\sigma_g}} \exp\left(-\frac{fitting_i(t)^2}{2\sigma_g^2}\right) \tag{3.9}$$

Using the distributions explained above, and assuming independence between them, the final weight of a particle is calculated as:

$$\pi_i(t) = P_{cd}(e_i^h(t))P_{cd}(e_i^t(t))\phi_i(t) \tag{3.10}$$

Equation 3.10 is able to manage uncertainty in the depth information. In the worst case (absence of information about disparity), the weight of a particle is based on color and gradient information uniquely. However, the greater the amount of disparity found, the greater its influence on the final particle weight. The final person position is assumed to be the mean of the state $\mathcal{E}[S(t)]$.

Assuming independence in Eq. 3.10 allows to speed up the particle computation. For each particle, the value $P_{cd}(e_i^h(t))$ is calculated first. If it has a low value, the final particle weight will also be low. Thus, computing time is saved by avoiding the calculation of $P_{cd}(e_i^t(t))$ and $\phi_i(t)$ when $P_{cd}(e_i^h(t))$ is sufficiently low.

Finally, the target color models $\hat{q}_{E^h}$ and $\hat{q}_{E^t}$ are updated at the end of each iteration step in order to adapt the tracking process to illumination changes. However, the target color models are only updated when the weight of the final estimated state $\pi_{\mathcal{E}[\mathcal{S}]}$ is above a certain threshold π_T in order to avoid including as part of the updated models elements from the background or from occluding objects. The target color models are updated as proposed by [NKMG03] using the projection of the 3D model indicated by $\mathcal{E}[S(t)]$.

3.1.2 Experimental Results

This section explains the experimentation carried out in order to validate this proposal. The recorded sequences show scenes with a varying number of people (from one up to four) interacting in a room. In the sequences, people perform several types of interactions: walk at different distances, shake hands, cross their paths, jump, run, embrace each other and even quickly swap their positions trying to confuse the system. People were instructed not to walk farther than 5 m from the camera. At larger distances the depth errors obtained became too high because of the narrow baseline of the SV system. A total of 7 different people participated in the tests.

Our experimentation aims to evaluate the tracking error in determining the 2D person head position in the reference camera image. In order to obtain quantitative measures of the tracking error, the people head position have been manually determined in each frame of the sequences. In total, there have been manually extracted 4460 positions from the sequences recorded.

In unimodal problems such as this, the final mean state $\mathcal{E}[S(t)]$ might be considered as the best person position estimation. Thus, the 2D tracking error is calculated as the distance from the manually determined position to the upper ellipse centre when the 3D model is projected from $\mathcal{E}[S(t)]$.

As previously indicated, the performance of methods based on PF increases as the number of particles grows. However, the higher the number of particles employed, the higher the computational time required for the algorithm is. Therefore, it is important to analyse the error of the tracker as a function of the number of particles in order to decide the most appropriate configuration for a particular application. Therefore, each sequence has been evaluated for an increasing number of particles. However, because of the stochastic nature of the algorithm, each test has been repeated several times with different seeds for the random number generator. In order to run the tests, the algorithm parameters have been experimentally determined as $\sigma_z = 0.1$, $\sigma_c = 0.2$ and $\sigma_g = 0.3$.

The analysis of Fig. 3.2 reveals that for a low number of particles, the algorithm obtains relatively high errors. However, there is a rapid improve-

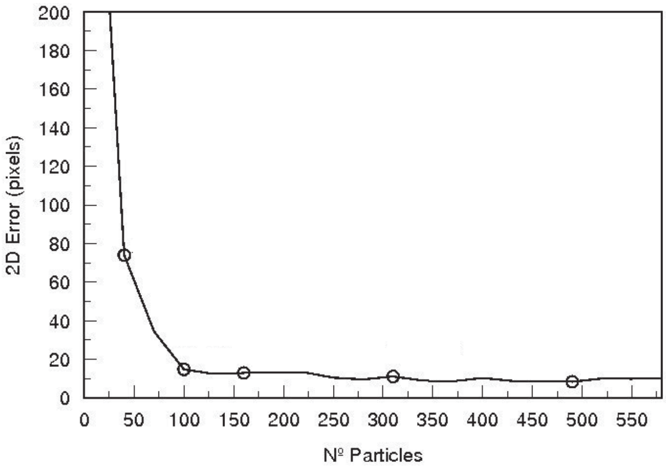


Figure 3.2: Tracking error in determining the 2D head position

ment of the performance as the number of particles grow up to the limit of 100 particles. As it can be noticed, no relevant improvements are achieved above this limit and the 2D tracking error was about 20 pixels. It also possible to conclude that the proposed method can be considered valid for real-time tracking purposes as the execution times per iteration was around 10 ms.

3.1.3 Summary and Final Remarks on the Probabilistic Approach

An approach to the person tracking problem based on combining multiple visual cues using a probabilistic particle filtering approach was presented. This method employs a 3D rigid human body model comprised by two ellipses: one for tracking the person head an another one for his/her the torso. Particles represent possible 3D positions for the model that are evaluated by examining their projection in the camera image. This method integrates depth, color and gradient information to perform a robust tracking.

Depth information cannot be always extracted because of occlusions or absence of texture. Our method is able to deal with this problem by defining a certainty measure that indicates the degree of confidence in depth information. The confidence measure is employed to modify the probability distribution function employed for weighting the particles. The greater is the amount of disparity found, the greater is its contribution to the final particles weights and vice versa. In the worst case (absence of information about disparity), the proposed algorithm makes use of the information available (color and gradient) to perform the tracking. The proposed algorithm does no only determines the 3D person position but also his/her head position in the camera image. This is a very valuable piece of information for Human-Computer Interaction (HCI) and Human-Robot Interaction (HRI) tasks (e.g., face pose estimation, expression analysis).

Several color-with-depth sequences have been employed in order to test the validity of our proposal. The sequences recorded show a varying number of people (from one up to four) interacting in a room. In the sequences, people perform different types of interactions: walk at different distances, shake hands, cross their paths, jump, run, embrace each other and even quickly swap their positions trying to confuse the system. The tracking errors have been calculated for different number of particles in order to determine the number of them that allows an appropriate trade-off between tracking error and computing time. The experimental results show that the proposed method is able to determine, in real-time, both the 3D position and the 2D head position in the camera image of a moving person despite of the presence of other people. Besides, the proposed method is able to deal with both partial and short-term total occlusion.

3.2 Possibilistic Approach for Human Detection and Tracking

After presenting the probabilistic approach for the tracking of people, a FL approach will be presented. This second proposal tries to overcome some situations which were not considered in the probabilistic approach. Firstly, and because in the approach presented in Section 3.1 the background information is not taken into account, there are some cases in which the background colour can be confused with the colour model of the tracked objects. In this second approach, foreground and background information are used which avoid certain confusing situations. Secondly, the approach mentioned in Section 3.1 does not handle certain occlusion situations which are now included in the “possibilistic” approach. Thirdly, in the probabilistic approach, there is only one confidence measure based on the disparity information. According to the amount of disparity, the algorithm computes the weight of a particle using uniquely colour and gradient information or also using depth information. On this Section’s approach, not only disparity information is taken into consideration for computing confidence levels, but also the distance at which the person is located and the possibility of being occluded. In addition to these advantages, a more elaborated people detection method is employed in order to reduce the amount of false positives. Furthermore, as we are going to see in Section 3.2.2, the 2D tracking error is better than on the method presented in Section 3.1.

The previously mentioned advantages are mainly due to a higher number of information cues in this section's approach. This is achieved with the help of FL which has the ability of managing an increasing amount of information, in a simple way. In addition, the use of FL to compute the final weight of each particle brings different benefits compared to the probabilistic approach. First, by using probability models to evaluate particles, it is assumed that variables follow a probability distribution. That is, uncertainty could be modelled in a probabilistic approach by modifying the probabilistic distribution function by means of some parameter. Those assumptions sometimes are not exactly true or are hard to be modelled. Nevertheless, with FL one can achieve the same goal in a more flexible way, without being restricted to particular aspects of the probability distributions. Secondly, FL easily allows to incrementally add other sources of information, in case our system needs so. By using linguistic variables and rules to express relationships the system becomes more understandable and similar to the way humans represent and deal with knowledge.

On the other hand, there is the drawback of using more information cues which is directly related to the computing time. This makes it even more important to correctly adjust the number of used particles, in order to find the best trade-off value between computing time and accuracy. In our case, and as it will be shown, the number of 50 particles allowed a good trade-off in the "possibilistic" approach.

3.2.1 Proposed Method

In this section, the process of people detection is first explained just before the description of the people tracker method takes place. FL is used in both phases and a new FLPF is used on the tracking phase.

People Detection

In this subsection, our possibilistic approach to detect new people in the surroundings of the robot is presented. The face detector tool, the concept of "projection of a person", the background extraction method as well as the occlusion handling technique are all explained in the next paragraphs.

Finally in the last part of this subsection, the algorithm to detect new people, which uses these mentioned techniques, is presented.

Face Detection

The people detection process begins with a face detector phase. The system employs the face detector provided by the OpenCV Library ([Sou10]). This face detector is also described in [BK08]. It is not in the scope of this work to develop face detection techniques since there is plenty of literature about it in [YKA02]. The face detector is based on Viola and Jones' method ([VJ01]) which was later improved by Lienhart and Maydt ([LM02]). The implementation is trained to detect frontal views of human faces and works on gray level images, although it can be trained to detect other perspectives of human faces (for instance, lateral views). This detector is free, fast and able to detect people with different morphological faces. Nevertheless, the problem of false positives should be taken into account, no matter the detector chosen for this job.

The classifier outputs the rectangular regions of the frontal faces detected in the camera reference image. Each detected face position is firstly compared to the position of each of those people which are already being tracked. If the difference between each of those positions, is higher than $distNewFace$, which was experimentally tuned, the system initiates a procedure which goal is to reject potential false positives and which is described in the last part of this subsection.

Projection of a Person

The concept of "projection of a person" has already been partially presented in Section 3.1. In this work, it is defined as the 2D region that both face and torso of a standing up and average size person, would occupy either in the reference and in the distance image, if his or her face were approximately in the same 3D location of the detected face. For instance, in the reference camera image, it can be interpreted as the equivalent of e^h and e^t together, which were defined in the Section 3.1.

Thus, to explain the concept of "projection of a person" the same assumptions made in Section 3.1, regarding various anthropomorphic features of a human being, are taken into consideration, namely the face and torso approximate size. We consider that a person face roughly fits inside a $20x30$ cm ellipse, and his or her torso fits in a $40x60$ cm ellipse. We also assume that those ellipses' centers are roughly separated by a distance of 45 cms.

By assuming these values, it is possible to extract from both reference and distance images, the regions occupied by both head and torso of a person. Knowing the intrinsic parameters of the camera, and with the help of the distance of the detected face centre to the camera (obtained using the stereo algorithm), it is possible to define two elliptical regions in the reference camera image, denoted by RP_{ri}, which are equivalent to e^h and e^t defined in the last Section. Analogously, two elliptical regions in the distance image, denoted by RP_{di}, which had not been yet defined in Section 3.1, are also defined. In our notation, RP stands for Region of Projection while ri stands for reference image and di for distance image. Fig.3.3 shows those regions for two different people both in the reference image ($RP_{ri}1$ and $RP_{ri}2$) and in the distance image ($RP_{di}1$ and $RP_{di}2$).

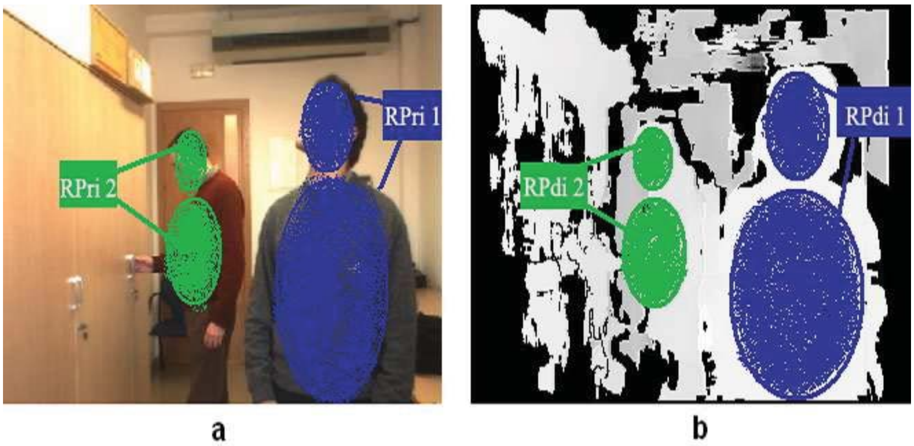


Figure 3.3: (a) Projection of 2 people on the reference image. (b) Projection of the 2 people on the distance image.

Background Extraction

The concepts of background extraction and occlusion map are used in the hereby presented approach. The first one consists in extracting the previously computed background of the environment in every new frame. To compute the background, an adjustable number of frames are used when the system is initialised. The background is modelled during the initialization of the system, by using information about the invariable colour and stereo data of the scene, as suggested by Harville in [HGW01]. Fig.3.4 exemplifies the background extraction method.

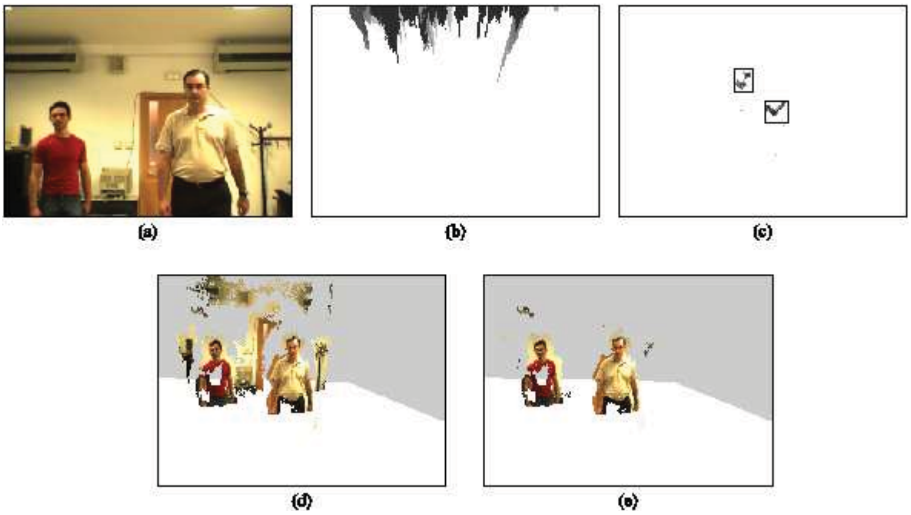


Figure 3.4: (a) Reference Image. (b) Background: Projection of detected static objects in the floor plane after the background extraction initial phase. (c) Projection of detected dynamic objects on the floor plane (corresponding to 2 people) during the experiments. (d) Reconstitution of the scene using distance information (static + dynamic objects). (e) Reconstitution of the scene using distance information and subtracting background.

Occlusion Map

In the proposed system stereo information of each detected or tracked person is stored in a vector variable. This information is updated in each frame. Therefore, it is possible to know who is closer or father regarding the camera and so to determine who is potentially occluding other people located behind.

The occlusion map for a frame is defined as a binary image where each pixel at each frame is set to 0 if it does not belong to a person. Each pixel at each frame is set to 1 whenever it is part of the mask of the person. This mask might be viewed as the silhouette of the person and is computed by using a flood fill algorithm starting from the centre of the person until it reaches its extremities. It is computed over a binary foreground image, which basically is an image where every pixel is set to 1 when it belongs to a foreground object detected for the current frame (for instance, a detected person) and to 0 when it does not belong to the foreground scene.

When the occlusion map is initialised, at every new processed frame, all pixels of this binary image are set to 0. This is done from the closest to the farthest located person so it is possible to compute, for each person, how many and which of his or her pixels are currently occluded by closer located people. In Fig.3.5, an example of this occlusion map is shown at the moment in which the algorithm is analysing if the occlusion map generated by the person which is located closer to the camera, interferes with the person which is farther from the camera. If there was, for instance, a third person in the image behind these two people, the algorithm would analyse whether the occlusion map generated by the two people which are closer to the camera (mask of the person which is closer to the camera plus mask of the second person closer to the camera) interfered with the third person located behind them. In Fig.3.5, on the right, it is possible to see how the person which is located closer to the camera occludes the one that is farther. In green, we represent the "projection of the person" which is located farther to the camera and it is possible to see that most of the pixels inside his "projection" are "covered" by those pixels which are part of the mask of the closest person.

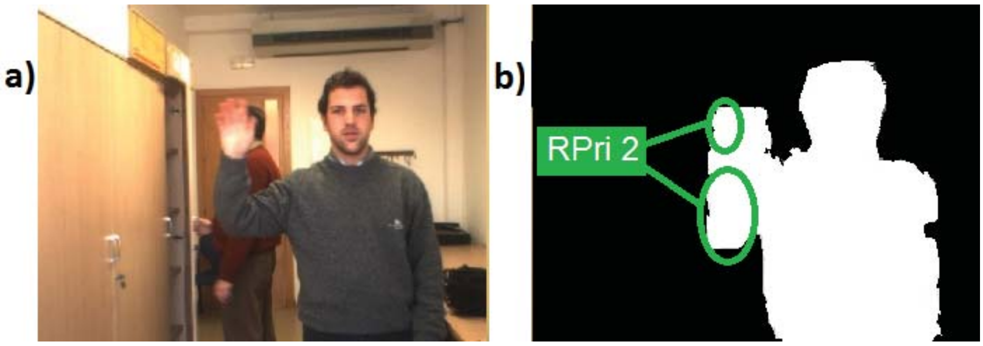


Figure 3.5: (a) Reference image with one person partially occluding another person. (b) Occlusion map where white pixels represent the pixels of the camera image occupied by the closest person to the camera. This way it is possible to know which pixels of the farther person are occluded.

During the people detection phase, the system knows which pixels were classified as being part of people in the previous frame, using the occlusion map. By knowing so it is able to determine if a candidate to new person potentially has a part of its body occluded. If so, these projections belong to a region where visual and depth information is not sufficient and consequently not reliable.

The occlusion map is also used in the tracking phase to compute a confidence level to the stereo and colour information. The methodology for using occlusion information in the tracking phase is explained in the next subsection.

First Test of People Detection

Regarding the next two Sections, the goal of these tests is to detect false positives, after detecting faces in the reference camera image. Let us call $RP_{ri}(DF)$ and $RP_{di}(DF)$ (where DF stands for Detected Face) to the projections of the person whose face belongs to a detected face, on both reference camera image and distance image.

The goal of the first test is to check whether inside $RP_{di}(DF)$ there are enough pixels respecting three conditions. First, they belong to the foreground (if they belong to the background they cannot be considered as being part of a person). Second, they have stereo information, ie, they are not unmatched points (if there is a person projected in $RP_{di}(DF)$ then this region should contain a high number of pixels with depth information). Third, they are not occluded. As people moving freely in the environment tend to occlude each other, if pixels inside $RP_{ri}(DF)$ and $RP_{di}(DF)$ are occluded, it is possible to infer that a person is fully or partially occluding other.

These three measures are fuzzified by three linguistic variables labeled as $ForegroundPixels$, $StereoPixels$ and $NonOccludedPixels$, respectively (see Fig.3.6). Using these three variables as input variables to the FS Test 1 (FST1) shown by Table 3.1, the fuzzy output $VisiblePerson$ is computed. FST1 and the rest of the FS shown in this work use the Mamdani inference method. The defuzzified value of $VisiblePerson$ indicates the possibility, from 0 to 1, whether region $RP_{ri}(DF)$ is likely to contain a visible person whose face is the one detected by the face detector. If this value is higher than α_1, the detected face passes to a second test.

At this stage, it is important to refer that all membership functions and rule bases were created using our expert knowledge and were then experimentally tuned. Rules which were considered irrelevant were eliminated and rules which were similar between themselves were merged.

Second Test of People Detection

The second test also checks whether $RP_{ri}(DF)$ may contain a true positive face. However the idea is different now. If there is a person in that region, then pixels inside $RP_{di}(DF)$ should have approximately the same depth as the centre of the face. In case the centre of the face is an unmatched point, the closest pixel which contains stereo information is used. Therefore the FS Test 2 (FST2) receives, as input, the difference between the average

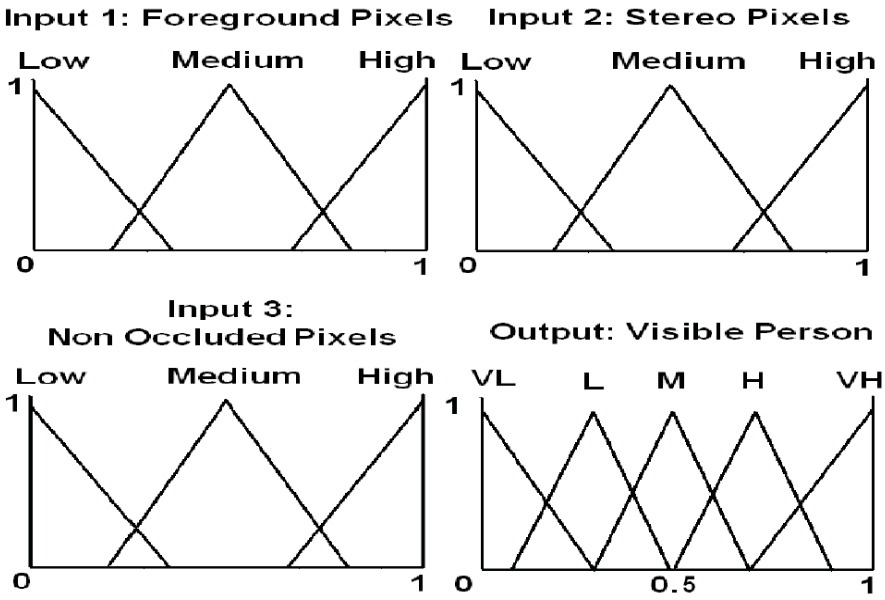


Figure 3.6: Fuzzy sets to assess detected faces with variables ForegroundPixels (ratio), StereoPixels (ratio), NonOccludedPixels (ratio) and output variable VisiblePerson

depth of $RP_{di}(DF)$ and the depth of the centre of the detected face as seen in Eq.3.11.

$$d = |Z(DF) - \frac{\sum_{j=1}^{n}(z_j)}{n}|. \tag{3.11}$$

where d is the difference to be computed, $Z(DF)$ the depth of the centre of the detected face (which is considered to be a good approximation of the face distance taking into account the precision of the SV camera and the expected algorithm precision), z_j the depth of the j pixel inside $RP_{di}(DF)$ and n the total number of pixels inside $RP_{di}(DF)$. This value is fuzzified by the linguistic variable $AverageDifference$.

FST2 also receives the standard deviation of the depth of those pixels belonging to $RP_{di}(DF)$, fuzzified by the linguistic variable $StandardDeviation$, and the depth at which the face was detected, fuzzified by the linguistic variable $Depth$. The depth of the detected face is used to compute the confidence that should be assigned to the values of the other variables. The farther the distance, the higher the uncertainty, according to the table provided by the manufacturer available online (see [Res05]). The output variable $SimilarDepth$ is computed by FST2 and its defuzzified value is a value between 0 and 1 corresponding to the possibility that $RP_{di}(DF)$ contains

Table 3.1: Rules for Fuzzy System Test 1.

IF			THEN
ForegroundPixels	StereoPixels	NonOccludedPixels	VisiblePerson
High	High	High	Very High
High	High	Medium	High
High	High	Low	Medium
High	Medium	High	High
...	...	...	...
Medium	High	High	Medium
Medium	High	Medium	Medium
Medium	High	Low	Low
...	...	...	...
Low	Medium	Low	Low
Low	Low	High	Very Low
Low	Low	Medium	Very Low
Low	Low	Low	Very Low

pixels with a depth value similar to the depth of the detected face. In Fig.3.7 linguistic variables $AverageDifference$, $StandardDeviation$, $Depth$ and $SimilarDepth$ (output) are shown. In Table 3.2 it is possible to find examples of the rules defined for FST2.

Finally, if this value is higher than α_2, it is assumed that a new person was detected and a new tracker is thus assigned to him or to her. The values for parameters α_1 and α_2 have been experimentally tuned. In our experiments, a value of $\alpha_1 = \alpha_2 = 0.6$ proved to be adequate in order to achieve a good performance.

The rules and linguistic variables defined for other FS are similar to the ones of Figures 3.6, 3.7 and Tables 3.1, 3.2 so that they are omitted in order not to be redundant.

People Tracking

In this subsection our approach to track people in the environment is presented. Firstly the Fuzzy Logic based Particle Filter (FLPF) is presented and then the Observation Model is introduced. Next the FS used in this work is presented in detail and finally, in the last paragraph of this subsection, the colour model update phase is presented so the system is able to handle changes in the colour model as illumination changes.

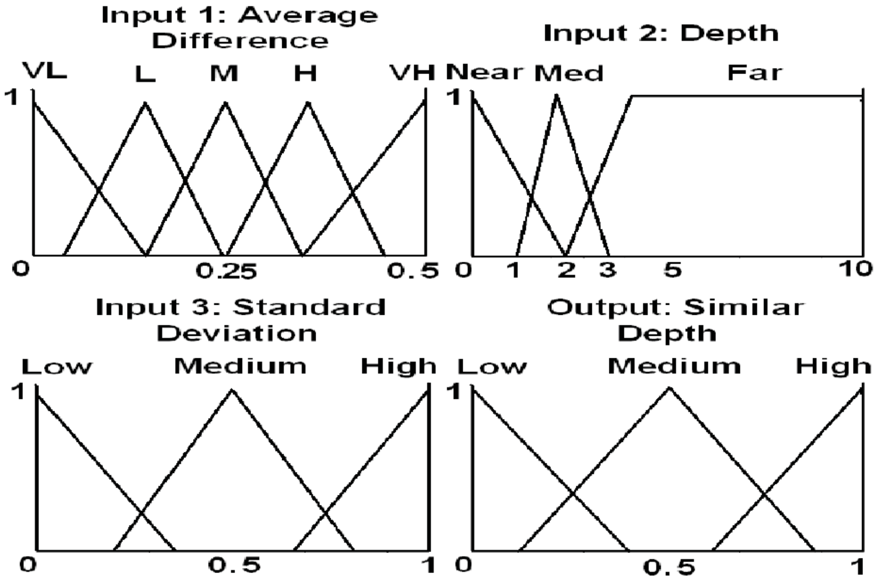


Figure 3.7: Fuzzy sets to assess detected faces with variables AverageDifference (meters), Depth (meters), StandardDeviation (meters) and output variable SimilarDepth

Fuzzy Logic based Particle Filter

In each frame, the tracking of people is done in depth order, which means that the closest person to the camera is firstly analysed until the one that is placed farther to the camera. There are as many trackers as people being tracked, and the maximum number of tracked people essentially depends on time processing constraints (one of the goals of this work is to comply with real time constraints) and on the amount of people that “fit” into the camera field of view. Considering the hardware of our system, this proposal allows up to 4 people to be tracked at the same time.

At the beginning of the tracking algorithm, and before the FLPF is integrally executed, a test is executed to assess the possibility that a previously detected face (by the face detector) corresponds to the face of the current person being tracked (see Fig.3.8(1)). To do so, the position, in the reference camera image, of the closest detected face $(CDF(t) = (x_{CDF}, y_{CDF}))$ to the previous position of the person being tracked $PersonPos(t-1)$ is selected. To consider that $CDF(t)$ corresponds to the new position of the person $PersonPos(t)$ it has to comply with two conditions. The first one is that its distance to $PersonPos(t-1)$ is less than an experimentally tuned threshold β. The second is that its evaluation value is above a certain γ

Table 3.2: Rules for Fuzzy System Test 2.

IF			THEN
AverageDifference	StandardDeviation	Depth	SimilarDepth
VL	Low	Far	High
VL	Low	Medium	High
VL	Low	Near	High
L	Medium	Far	Medium
...	...	...	...
M	Medium	Far	Low
M	Medium	Medium	Medium
M	Medium	Close	Medium
...	...	...	...
VH	Medium	Close	Low
VH	High	Far	Low
VH	High	Medium	Low
VH	High	Close	Low

threshold, which once again was experimentally tuned and set to 0.8. This evaluation method is described in the next subsection. In the case that the particle complies with these two conditions, $CDF(t)$ is considered to be the new position of the person $PersonPos(t)$. The aim of this procedure is to avoid all the particle filtering process, when there are strong suspicions that some specific face could be the face of the person that it is being tracked. By adopting this procedure, extra processing time is spared and the algorithm tracking accuracy is improved.

When no face is detected in the "neighbourhood" of the last position of the tracked person, the FLPF takes place (Fig.3.8(2)). PF can estimate the state of a dynamic system $PersonPos(t)$ from sequential observation $z(t)$. The variable $PersonPos(t)$ is defined as the position x_p, y_p of the centre of the person face. To achieve that estimation $\mathcal{E}_t$, a weighted set of J particles $S(t) = \{(s_i(t), \pi_i(t))\}$, with $i = 1..J$, is computed, where $s_i(t) = (x_{si}, y_{si})$ represents a possible state of the system, and $\pi_i(t)$ is a non-negative numerical factor called importance weight which represents an estimation of the observation density $p(z(t)|s_i(t))$. Our approach is based on the typical structure of the Condensation algorithm ([IB98]), which is partially adapted with new concepts that will be described. In our system, $\pi_i(t)$ is not computed by means of probabilistic assumptions but using FL. This is achieved by combining the output of several hierarchically connected FS.

The value of J was experimentally tuned to 50, as lower values might compromise accuracy and higher values might compromise processing time (and real time constraints whenever there are several people being tracked). As referred before, when no face is detected in the "neighbourhood" of the tracked person last position, the algorithm uses the previous position of the person $PersonPos(t-1)$ to create a set of particles $S(t)$. The propagation model of the particles is based on the previous position of the person plus some δ random Gaussian noise with parameters $N(\mu = 0\ px, \sigma = 30\ px)$. The idea is to generate most particles in the surroundings of the previous position and only a few farther, as people are not expected to move fast from frame to frame. The new samples $s_i(t)$ are then weighted.

The weight $\pi_i(t)$ of each particle is computed, taking into consideration the new observations obtained from the FS.

Observation Models

After generating the set of particles, the process of evaluating the possibility $\pi_i(t)$ that each particle corresponds to the tracked person (Fig.3.8(3)) begins. The observation model for each particle is based on the output of different FS as shown in Fig.3.9. There are 5 FS, which are called FSRI (FS Region Information), FSFI (FS Face Information), FSC (FS Confidence), FSPPDI, (FS Particle to Person Distance Information) and FSTI (FS Torso Information). They are sorted out according the type of information which each of their variables represent. The whole system is structured in a hierarchical way, which is one alternative presented in the literature ([Tor02]) to overcome the problem of reducing the complexity of rule understanding, when several variables are used in FS. Therefore, a two layer FS approach is used, which takes into account the confidence level of the outputs of some of the FS. The overall result for each particle is given by $\pi_i(t) = OutFSC * OutFSPPDI * OutFSTI$ where each parcel corresponds to the defuzzified output of a FS and is a value between 0 and 1 (see Fig.3.9).

The new position of the tracked person, $PersonPos(t)$, is equal to the final state estimation $\mathcal{E}_t = \mathcal{E}[S(t)]$ which is obtained from the mean of the state $S(t)$ by weighting all particles $s_i(t)$ (see Fig.3.8(4)).

For a better understanding of our algorithm, a detailed description about the functioning of the FLPF algorithm is shown in Fig. 3.8.

Fuzzy Systems Description

In the next paragraphs it will be described each of the FS used to compute the value of $\pi_i(t)$. Because of the similarity between FS, all labels and

```
1. Evaluate whether there is a Detected Face with high possibility of being
   the face of the person being tracked:
   $CDF(t) = (x_{CDF}, y_{CDF})$ where $CDF$ stands for Closest Detected Face
   IF $(CDF(t) - PersonPos(t-1) < \beta)$ AND $(\pi_0(t) > \gamma)$ THEN
   $PersonPos(t) = CDF(t)$ (Go to Step 5)
   ELSE

2. Compute a sample set $S(t)$ from $PersonPos(t-1)$ as:
   Set $s_i(t) = PersonPos(t-1) + N(0,1)$ with $i = 1..J$

3. Measure and weight each sample in terms of the new observation:
   $\pi_i(t) =$ OutFSC*OutFSPPDI*OutFSTI
   Then normalize so that $\sum_{i=1}^{J} \pi_i(t) = 1$

4. Estimate the new state $S(t)$ and calculate its weight:
   $\mathcal{E}_t = \mathcal{E}[S(t)] = \sum_{i=1}^{J} \pi_i(t) s_i(t)$.

5. Update the occlusion map
```

Figure 3.8: Algorithm employed for tracking each person

rule bases of each FS are not described. However, by taking a look at the examples presented on the previous sub section, the reader may have a good idea about the type of values used in these FS .

In the evaluation process of $S(t)$ the concept of "projection of a person" is also used. In this case, the position (x_{si}, y_{si}) of the particle currently being evaluated is used as the centre of the face to compute the projection of the person $RP_{ri}(s_i)$ in the reference camera image and $RP_{di}(s_i)$ in the distance camera image.

The goal of FSRI is to evaluate the region $RP_{di}(s_i)$ (see Fig.3.3). This evaluation takes into consideration only aspects related to the possibility that some object, similar to a person, is projected in that region. The first step is to compute the area of $RP_{di}(s_i)$. After obtaining this information three linguistic variables that are called $ForegroundPixels'$, $StereoPixels'$ and $AverageDeviationDifference$ are used. $ForegroundPixels'$ and $StereoPixels'$ are defined in a similar way to $ForegroundPixels$, $StereoPixels$ of the previous subsection. $AverageDeviationDifference$ provides information about the difference between the depth of s_i and the average depth of all pixels inside $RP_{di}(s_i)$. This value is also fused with the standard deviation of the depth of those pixels. The reason for defining this variable is that, all pixels inside $RP_{di}(s_i)$, should have approximately the same depth

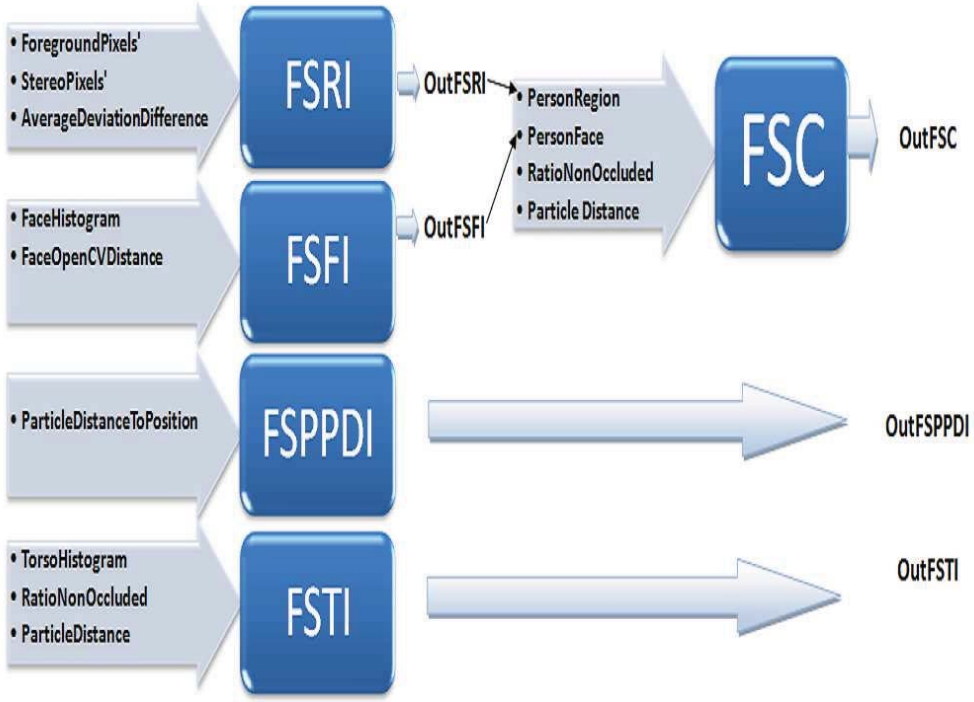


Figure 3.9: Fuzzy Systems used to evaluate de overall quality of each generated particle. For each FS, the input linguistic variables are specified.

as s_i and should have approximately the same depth between them, as long as they belong to some person or object. We would like to highlight the fact that only pixels that are considered as not being occluded by other person are taken into account. To know which pixels are in this situation, the occlusion map is used. The occlusion map is updated at the end of the tracking cycle, for each tracker. These values are the input to FSRI that outputs a deffuzified value between 0 and 1. The higher amount of foreground, stereo pixels and lower difference in average and standard deviation, the closer the output is to 1. A value closer to 1 means that, in the area represented by $RP_{ri}(s_i)$, it is likely to have some object that could hypothetically be a person.

The scope of FSFI is to evaluate face issues related to the person being tracked. The first step is to define two linguistic variables called *FaceHistogram* and *FaceOpenCVDistance*. The first one contains information about the similarity between the face region of $RP_{ri}(s_i)$ and the face histogram of the person being tracked. As people, from frame to frame (at a 15 fps frame rate), do not tend to move or rotate their face so abruptly, those histograms should be similar. The elliptical region of the face is used to create a colour model ([Bir98]). The difference between the face histogram of region of $RP_{ri}(s_i)$ and the face histogram of the person being tracked is measured. This difference is based on a popular measure between two colour distributions: the Bhattacharyya coefficient as explained in Section 2.2.2. Once again, only pixels that are not occluded are used in this process. This method gives the similarity measure of two colour models in the range $[0, 1]$. Values close to 1 mean that both colour models are identical. Values near 0 indicate that the distributions are different. An important feature of this method is that two colour models can be compared even if they have been created using a different number of pixels. The second linguistic variable measures the distance between s_i and the position of the nearest face to s_i detected by the OpenCV face detector. Although OpenCV is not 100% accurate, most of the time this information can be worth as it can tell if there is really a face near s_i. The deffuzified output of this FS is also a number between 0 and 1 where 1 is an optimal value.

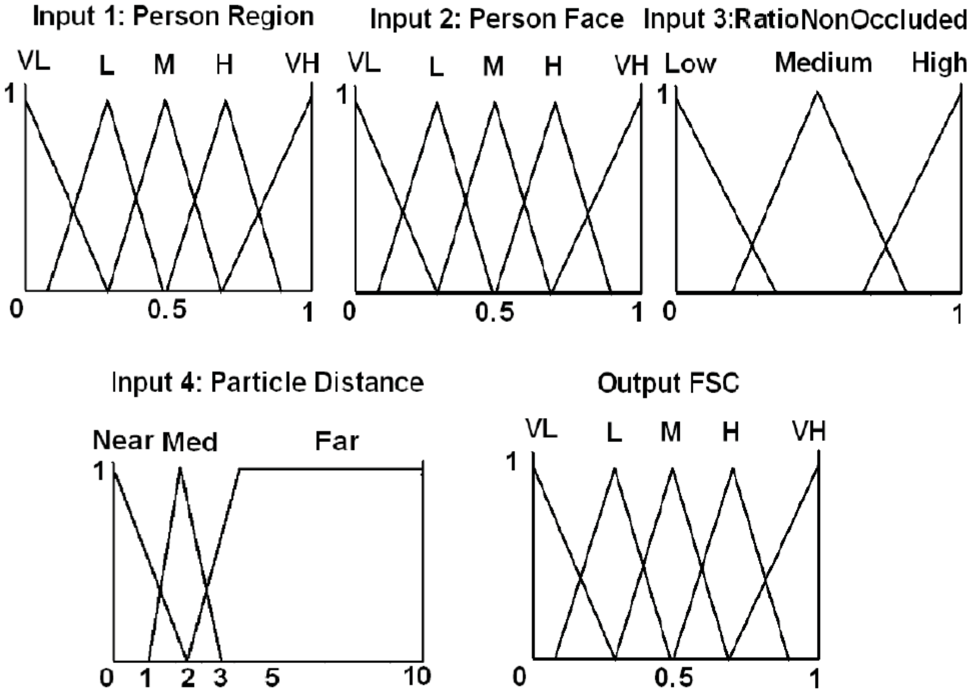


Figure 3.10: Variables for fuzzy system FSC, PersonRegion, PersonFace, RatioNonOccluded, ParticleDistance (meters) and OutputFSC

The deffuzified outputs of FSRI and FSFI are then provided as input of FSC. The aim of this FS is to measure the confidence of the outputs of FSRI and FSFI based on occlusion and depth information. As including new variables in FSRI and FSFI would make it more difficult to define rules and better understand the whole system, a Hierarchical Fuzzy System(s) (HFS) structure was chosen, allowing to measure the confidence of the mentioned outputs. Thus, for FSC, four linguistic variables called $PersonRegion$, $PersonFace$, *RatioNonOccluded* and *ParticleDistance* are defined in order to compute its final output as it is possible to see in Fig.3.10. $PersonRegion$ and $PersonFace$ have five linguistic labels Very Low, Low, Medium, High and Very High distributed in a uniform way into the interval $[0, 1]$ and its inputs are the defuzzified outputs of FSRI and FSFI respectively. *RatioNonOccluded* contains information about the ratio of non occluded pixels inside $RP_{ri}(s_i)$. The higher the number of non occluded pixels, the more confidence on the output values. In other words, the more pixels from $RP_{ri}(s_i)$ and $RP_{di}(s_i)$ which can be used to compute foreground, depth, average information and histogram, the more confidence on the outputs of FSRI and FSFI. Finally *ParticleDistance* has information about the distance of the evaluated particle (s_i). As errors in stereo information increase with distance, the farther the particle is located, the less trustable it is in means of depth information. The defuzzified output of FSC ($OutFSC$) is also a number between 0 and 1. Higher values indicate a region with higher possibility to contain a person. Rules for this FS can be seen in Table 3.3.

Table 3.3: Rules for Fuzzy System Confidence (FSC).

IF				THEN
PersonRegion	PersonFace	RatioNonOcc	ParticleDistance	Output FSC
VH	VH	High	Close	VH
VH	VH	High	Medium	VH
VH	VH	High	Far	H
VH	VH	Medium	Close	H
...	...	...	...	...
M	M	High	Close	M
M	M	High	Medium	M
M	M	High	Far	L
...	...	...	...	...
VL	M	Medium	Close	L
VL	M	Medium	Medium	VL
VL	M	Medium	Far	VL
VL	M	Low	Close	VL
VL	L	Low	Medium	VL
VL	VL	Low	Far	VL

With respect to FSPPDI, its goal is to evaluate whether s_i is likely to be the person being followed, by taking into consideration the distance to the previous location of the person (in the frame before). Due to the frame rate used, people from frame to frame are not expected to move significantly. Therefore, only one variable called $ParticleDistanceToPosition$ is defined, which contains information about the distance in pixels between the position of s_i and the position of the currently tracked person ($PersonPos(t-1)$). The deffuzified output is, once again, a value between 0 and 1 represented by $OutFSPPDI$. An output equal to 1 means that s_i is located exactly in the same place where $PersonPos(t-1)$ was located.

The last FS, FSTI is related with torso information. Identically to FSFI, a variable that translates the similarity between the torso histogram information of $RP_{ri}(s_i)$ and the histogram information of the torso of the person being tracked is defined. This variable is called $TorsoHistogram$. Similarly to FSFI, only pixels that are considered as non occluded are used. For this FS, the variables $RatioNonOccluded$ and $ParticleDistance$ are also defined analogously to FSC. This way, we are adding a measure of confidence for the output which, after its deffuzification, is called $OutFSTI$ and has a value between 0 and 1.

As said before, all these outputs are multiplied and the result is a value between 0 and 1. Then, a weighted average of the position in the reference image $PersonPos(t)$ is computed, by taking into consideration all the possibility values for the set of particles. A particle that has a possibility value closer to 1 weights much more than one with a possibility value of 0. Its region of projection is also added to the occlusion map, so the following trackers and the people detection algorithm know that there is already a person occupying that region.

Model Update

Changes in the illumination conditions and person different perspectives might alter the observed colour distribution of the tracked region. Therefore, it is necessary to update the head and torso colour models to achieve robust tracking. For that purpose, after the tracking process is concluded, the projection of the person on the reference camera image RP_{ri} is used to update his or her colour model. The pixels of RP_{ri} are employed for creating the new observed colour model as:

$$\hat{q}_E(t) = (1-\alpha)\hat{q}_E(t-1) + \alpha\hat{q}_E a(t) \tag{3.12}$$

where the parameter $\hat{q}_E a(t)$ refers to the observed colour model for the current estimated projection and $\alpha \in [0, 1]$ determines the contribution of the observed colour model to the updated target model. In order to avoid the inclusion of pixels from the background or from occluding objects as part of the updated model, only pixels that are part of the foreground, that are not occluded, and belong to RP_{ri} are employed. Finally, we have opted to set $\alpha = 1 - \rho(\hat{q}_E(t), \hat{q}_E a(t))$. In that way, the model is automatically updated accordingly to its difference to the actual observed colour model. The higher the difference between them, the higher the value employed for α. This is done both for the head and torso colour models independently.

3.2.2 Experimental Results

Previously, we have mentioned several advantages of using this approach regarding the method presented in Section 3.1, after taking into consideration other information cues that were not used in the first approach. Nevertheless, it is necessary to validate this proposal by comparing it to well known methods in this field of research. This experimental study is shown in this Section.

The achieved operation frequency of our system depends on the number of people being tracked and the number of particles that were used by the PF algorithm. As each tracked person implies a new tracker, processing time increases for each added tracker. Up to 4 people are allowed in order to the system to be able to perform in real time with our hardware.

For this experimentation, different colour-with-depth sequences have been recorded using our stereo camera. Videos were recorded in different rooms with different lightning conditions so a diversity regarding background scenarios was taken into consideration. Several people participated in the recording and they were instructed to move freely and to simulate different interaction situations either with other people or with the camera.

The aim was to check whether our algorithm was able to keep track of different people in several situations that are part of the daily life of a human. The hereby presented algorithm was compared with an adaptation of Nummiaro's algorithm ([NKMG03]) which is a PF approach that uses the Bhattacharyya coefficient to compare two colour regions. A comparison with the Kalman/meanshift tracker proposed by [CR00], which is implemented in the OpenCV library, was also done. This version of the Comaniciu's algorithm is able to track only one person at a time. Nevertheless this feature is enough for testing accuracy and executing times, which are our main concerns.

The comparison of the proposed approach with the Comaniciu's and the Nummiaro's based algorithms is made by measuring the distance error between the indicated position provided by all the three algorithms and the manually defined position, on the reference image, of the person being tracked. The error concerning the size of the indicated face rectangle was also measured. The projected size of the face in the camera image and the manually defined size of the face was also compared. To do so, both differences between the equivalent sides of both rectangles were used. For approaches that output an ellipse, the longest axis of the ellipse is considered equivalent to the longest size of the manually determined rectangle, and the shortest axis of the ellipse is considered equivalent to the shortest size of the manually determined rectangle.

In total, more than 5000 frames have been manually annotated. These frames correspond to 8 videos that last between 40 and 60 seconds each. Statistical information indicating the error values for different algorithms are presented. The RMSE (Root Mean Square Error) between the manually determined positions and the position indicated by the tracker as well as the RMSE between the manually determined rectangle sizes of the faces and the ones indicated by the trackers were chosen as error measures. Please note that because of the stochastic nature of the PF algorithms used, results are affected by the initialization of the random number generator. To avoid this problem, each experiment has been repeated 30 times with different initialization seeds. The values of the mean values of the RMSE for the set of frames concerning the 30 runs are presented on Table 3.4.

Concerning the processing time, an average of 22 ms was achieved for one for each tracking cycle when using the 50 particle version of the algorithm. Despite the high amount of data involved, this time proves that the hereby presented algorithm can be used in real time environments while achieving a more accurate and robust tracking than other traditional algorithms.

By looking at Table 3.4 it is possible to see that, once again, our algorithm outperforms other algorithms in accuracy, without compromising real time performance. Comparing the 2D tracking error of our two tracking approaches, we can see that the second algorithm obtains 8.85 px, as error value, against almost 20 px of the first algorithm. The error is better in the second algorithm but the execution time has increased regarding the first algorithm.

Depending on the colour of the target and the background, other al-

Table 3.4: Comparison between approaches

	Our Approach	Nummiaro Based	Comaniciu Based
RMSE Position	8.85 px	35.99 px	58.49 px
RMSE Rectangle Size	4.88 px	61.39 px	220.87 px
Processing Time per cycle and person	22.64 ms	12.62 ms	17.65 ms

gorithms can vary their accuracy while our algorithm generally does not lose track of its targets. Please note that these methods, which are based only on colour, perform very poorly when the background of the scene presents a colour model very similar to the colour of the skin or clothes. In those cases, the algorithm simply does not work, sometimes detecting the whole background and/or image as the initial person being tracked. In Fig.3.11 (b) and (c) and specially in Fig.3.12 (b) and (c) examples that illustrate these remarks can be found.

By taking a deeper look at Fig.3.11 and Fig.3.12, it is possible to observe different aspects regarding the tested algorithms. Fig.3.11 represents a scene with two people, slightly moving forward and backward, partially or totally occluding their face. Fig.3.12 helps to understand how using only colour information on a tracking algorithm approach, can critically downgrade the accuracy of those algorithms. Five frames from each video were chosen in order to exemplify how different algorithms track both of them.

In Fig.3.11(a) it is possible to observe that the proposed algorithm manages to keep track of those two people without ever losing their track. In frames number 205 and 275, sometimes, the reader may see that the square of the face is not totally centred, but the error is not substantial. This was one of our main goals, ie, to acquire a reasonable approximation of each face region. By observing Fig.3.12(a) one may see that our algorithm keeps track of people in different situations, even when they cross their paths. Below, we will analyse it deeper in Fig.3.13.

If confidence is put only on colour information, as exemplified in Fig.3.11 (b) and Fig.3.12 (b) it is very common that the tracking algorithm starts to assume that neighbor regions are part of the head and starts to slide from the target person (Fig.3.12(b) frame 185 on both people) to similar colour objects. In (Fig.3.12(b)), it can be observed that the algorithm also loses its target. The new squares observed in frames 343 and 399 correspond to new trackers, as the system detects faces that are faraway from the previously tracked people.

In Fig.3.11(c), although only one person is detected in the available version of the Comaniciu's algorithm, it is easy to observe that this algorithm works quite well, although there are some issues that should be pointed out. Indeed, it is possible to see that the Comaniciu's algorithm works fine in this scene but, and because it makes the tracking based on the skin colour model of the face, when a person turns back towards the camera, it detects his neck as the whole head area. Furthermore, when a person is facing the camera, it includes the neck as part of the person face. This aspect could turn out to be a problem, for applications that make use of face features. Concerning Fig.3.12(c), as it can be observed, background presents a very similar to face colour model which turns out to be a reason for Comaniciu's algorithm to rapidly lose its tracked person. In this kind of situation the algorithm fails completely.

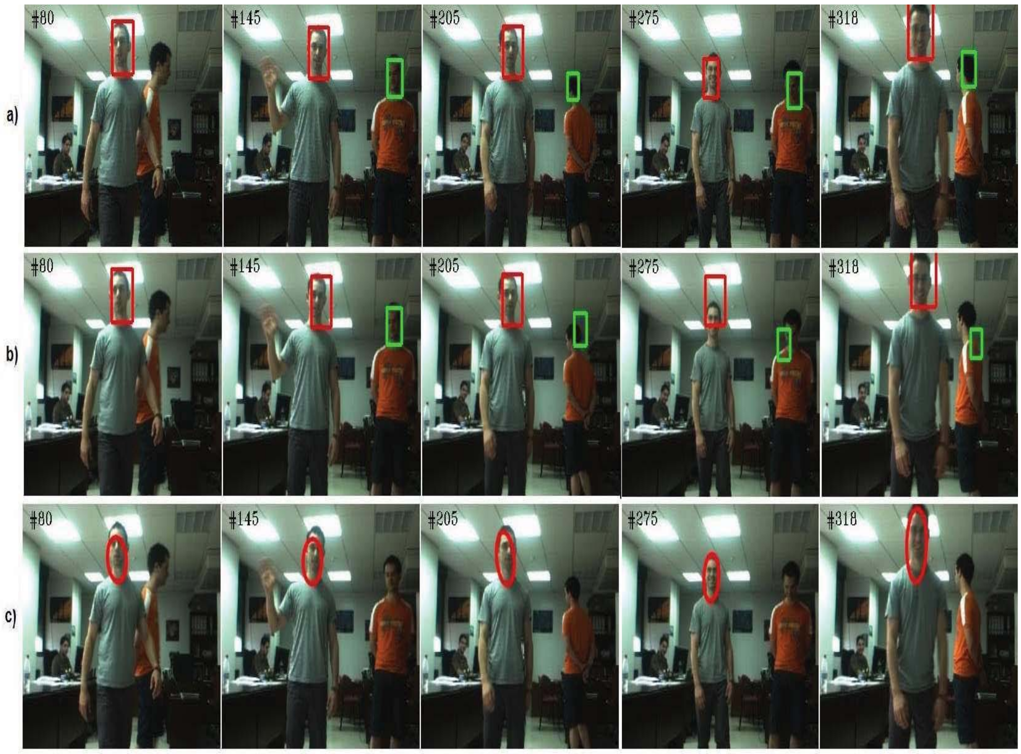


Figure 3.11: a) Proposed Algorithm; b) Nummiaro Based Algorithm; c) Comaniciu Algorithm.

Despite the good results achieved by our proposal, there might exist scenarios where, when two people dressed with the same colours are located near to each other, the system momentarily loses track of the tracked people. Nevertheless, this hypothetical scenario affects all the analysed algorithms in this section. This issue can be solved in a near future by providing more information sources to the system which, in the presented approach, should be as simple as adding new FS or rules to the existing ones. This issue might be considered as an important advantage with respect to other approaches.

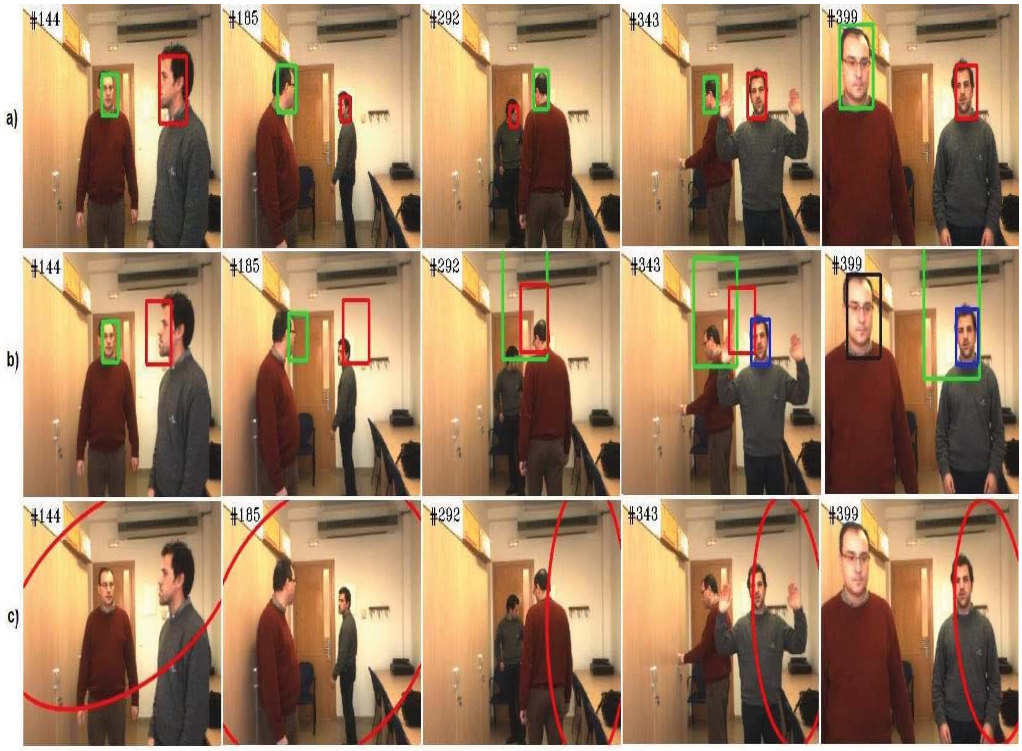


Figure 3.12: a) Proposed Algorithm; b) Nummiaro Based Algorithm; c) Comaniciu Algorithm.

Behavior results with two people interacting

Finally, in Fig.3.13 there is an example of four frames taken from one of those videos, with both reference image and distance image shown for each frame. The aim of this example is to show how our algorithm behaves during a natural interaction between 2 people. In the distance image, lighter areas represent shorter distances to the camera. In Fig.3.13(a) it is possible to see that the system detected person A (square 1) but person B was not detected (due to the fact that the employed face detector only detects frontal faces). In Fig.3.13(b) the reader can see that person B was detected (square 2) as his head was now facing the camera. It is important to have in mind that the stereo camera sometimes produces errors that tend to decrease the accuracy of the stereo part of the algorithm. For instance, in the distance image, it may happen that the region of the face has the double of its real size. In Fig.3.13(c) it is possible to see that the size of the head of the person A, in the distance image, is much bigger that its actual size. In this experiment, people cross their trajectories achieving similar values for their positions. However, the system could still keep an accurate track for each of the people. The reason for achieving this accuracy relies on colour information that compensated the similarity of position information. Finally in Fig.3.13(d) it is possible to see that, for person A, although part of his body was occluded, the system could still achieve an accurate tracking, based on stereo information rather than colour information.

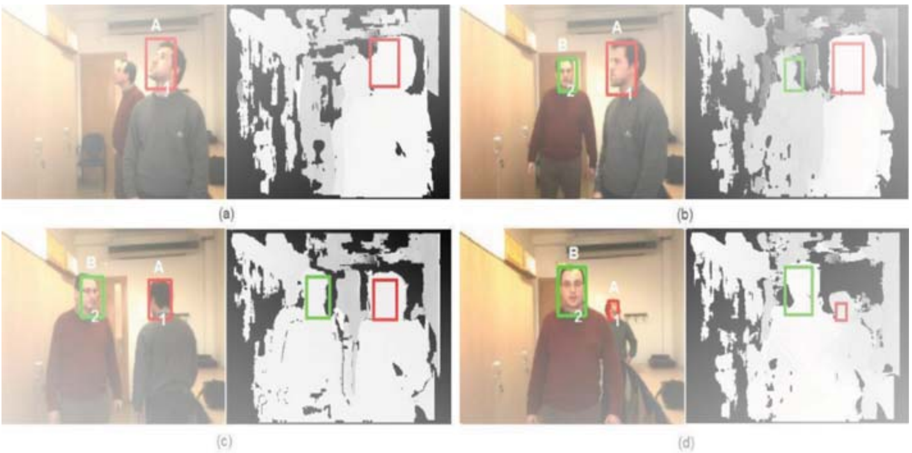


Figure 3.13: Different frames taken from a video with 2 people being tracked

3.2.3 Summary and Final Remarks on the Possibilistic Approach

A system able to detect and track various people simultaneously, using a new approach based on both colour and stereo information handled by means of FL, has been presented. The results showed that our system managed to keep track of people, in the reference image, in most of the situations where other trackers fail. It was tested in simulated complex real life situations, where people were interacting freely and occluding each other sometimes.

The method proved to be fast enough for detecting and tracking people simultaneously and therefore adequate to be used in real time applications.

The system uses FL in order to integrate information to detect and track people, managing the vagueness of the data provided by the sensors. FL is an interesting tool that has a proved efficacy for treating uncertain and vague information as well as noisy data from different sources. A modified PF is used to generate particles that are evaluated using FL instead of probabilistic methods. As it is known, information supplied by sensors is commonly affected by errors, and therefore the use of FS help to deal with this problem. By setting up linguistic variables and rules that deal with this problem, we achieved an efficient way of solving it.

Both FS used for people detection and the HFS used for the tracking process, deal with several sources of information as colour, position in the reference image, depth, occlusion and other data obtained from the SV. In this sense, information regarding depth and occlusion is used to create confidence levels to fuse, in an appropriate way, both colour and stereo information. Furthermore, the advantage of using several sources of information relies on the fact that these sources complement each other. Thus, when information about the position of people is not enough to identify them, colour as well as other information sources can be used to identify them. On the other hand, when the colour information extracted from a person is similar to the colour information extracted from another person or similar to the colour of the background, the stereo data is useful to distinguish between them. Overall, the people detection and tracking processes achieve very good results thanks to the fusion of these kinds of information.

Also, when FS are used to represent knowledge, the complexity in understanding the system is substantially lower, as this kind of knowledge representation is similar to the way the human being uses to represent its own knowledge. Furthermore, it allows an easy way of adding new features, just by adding more variables or FS. Thus, it will be easy to expand the system in the future, when new sources of information are available.

Chapter 4

Interest Detection and Attention Request

As the main motivation of this thesis is to improve Human-Robot Interaction (HRI), the work presented in this Chapter appears as a fundamental part after achieving the goal of detecting and tracking people described in the last Chapter. As a matter of fact, once the people are located, the level of interest of each person to interact with the robot is calculated by analysing his/her position and his/her degree of attention, as presented in Section 4.1. The position of a person is analysed using both his/her distance to the centre of the robot and his/her angle in respect to the heading direction of the robot. With respect to the degree of attention, this is determined detecting the orientation of the head, i.e., a higher degree of attention can be assumed when a person is looking at the system than when it is backwards. This analysis is solved by a view based approach using Support Vector Machines (SVM) which is briefly described in Chapter 2.3.2. Thanks to SVM, head pose can be detected achieving a great percentage of success independently of the morphological features of the heads. Fusing this information with Fuzzy Logic (FL), a level of interest in interacting with the robot can be computed for each detected person.

When the level of interest is high, the person is analysed in more detail to detect some of the interaction situations commented above. This is translated by analysing two typical interaction situation, arms movement and head shaking/nodding detection, as presented in Section 4.2. The presented approaches are not only valid for robotic applications but also in ambience intelligence that use stereoscopic devices.

4.1 Interest detection

This section explains our FL approach for estimating the interest of the detected people in interacting with the robot. The approach presented in this work is based on Stereo Vision (SV) but the system can be easily expanded to merge other sources of information. In Section 2.4, the general advantages of using FL in controllers were shown. In Section 3.2.1 the advantages of using FL in a tracking algorithm were shown. In the case of interest detection, these advantages also play a key role in the approach. This is to say, firstly, the robot has to deal with information from the SV system that is affected by uncertainty and vagueness. FL is a good tool to manage uncertainty using linguistic variables. Secondly, the human knowledge can be usually expressed as rules. FL allows to establish relationships among the variables of a problem through fuzzy rules providing an inference mechanism. Finally, there are methods in FL to fuse the results from several fuzzy rules in order to achieve a final overall result. Therefore, the system designed in this work, based exclusively on stereo information, could be easily integrated with other Fuzzy System(s) (FS) using other types of information as source sound localization, gesture analysis or speech recognition systems. In this work, the determination of the degree of interest of a person is based on its position and its degree of attention. The position of a person is analysed using both its distance to the center of the robot and its angle respect to the heading direction of the robot. The first feature is measured by the linguistic variable *Distance* and the second one by the linguistic variable *Angle*. Each one of these linguistic variables has three possible values as shown in Fig. 4.2. These two variables are used to establish the following rule: if the person is detected near the robot and more or less centred with respect to it, then it is assumed that the person is more interested in establishing an interaction with the robot rather than when the person is far or on the left or right side of the robot. Nevertheless, the position of the person is not enough to determine his/her interest in interacting with the robot. Thus, the third feature shown in this work is the person attention detected by the analysis of the head pose. To detect the head pose a view based approach using SVM has been employed. This approach will be now explained.

4.1.1 Estimating face attention using Principal Component Analysis and Support Vector Machines

One of the most prominent cues to detect if a person is paying attention to the system is the orientation of the face, i.e., a higher degree of attention can be assumed when a person is looking at the system than when it is backwards. This section describes our approach for face attention estimation.

Head poses have been classified in three main categories: "A" that comprehends all the frontal faces (faces looking directly at the camera), "B" that comprehends all the slightly sided faces (faces looking to some point slightly above, below or aside from the camera) and "C" that comprehends all the other faces (side faces, faces looking at some point in the ceiling or ground, backward heads). Figure 4.1 shows examples of each one of the categories employed.

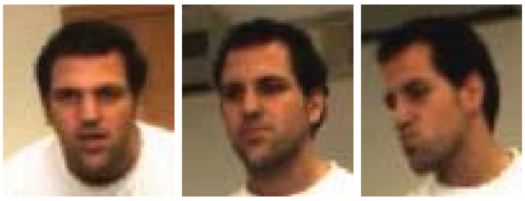


Figure 4.1: Head Pose Estimation: Classes A, B and C.

A head pose database comprised by a total of 4000 samples equally distributed among the three classes has been created. The database contains images of 21 different people (men and women), of different races, with different hair cuts and some of them wearing glasses. The database samples were manually classified into categories "A", "B" or "C" according to where people were looking at. All the images are gray-scale and 48x40 sized.

In order to reduce the data dimensionality, Principal Component Analysis (PCA) has been applied (see Chapter 2.3.1 for more information about PCA). A key aspect in PCA consists in deciding how many principal components are appropriate for a proper training. A low number of components reduces the computing time required, but also reduces the accuracy since part of the information is discarded, and vice versa. Therefore, tests with different number of components have been performed and the amount of 50 components has been chosen as it allows a good trade-off between classification accuracy and computing time.

The training process has been carried out using SVM. To certificate that results were satisfactory before applying the model 85%, of the data

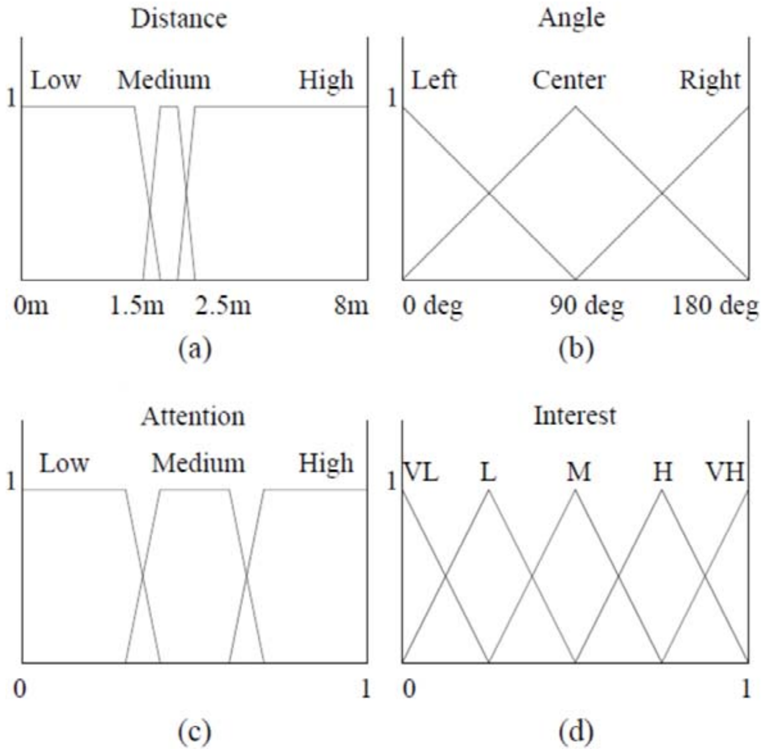


Figure 4.2: Fuzzy sets of the linguistic variables: (a) Distance (b) Angle (c) Attention (d) Interest

set has been used to train the SVM and the remainder 15% to test the model generated. The result on the test set was of 93.14% of accuracy.

For each detected person the SVM classifier estimates in real time the head pose in one of the three previously indicated categories. The output of the SVM classifier is translated into a numerical value by the definition of the variable $SVMOut$. The value of $SVMOut$ in the time t is

$$SVMOut_t = \begin{cases} 1 & \text{if output SVM = “A”;} \\ 0.5 & \text{if output SVM = “B”;} \\ 0 & \text{if output SVM = “C”.} \end{cases} \tag{4.1}$$

However, $SVMOut_t$ is an instantaneous value that does not take into account past observations. In order to consider past observations, the variable $HP_{(t)}$ is defined as:

$$HP_{(t)} = \alpha HP_{(t-1)} + (1-\alpha) SVMOut_t \tag{4.2}$$

where the initial value for HP is the first value of $SVMOut$ when the person is detected and α is a weighting factor that ponders the influence of past observations.

To deal with the uncertainty and vagueness in this process a linguistic variable called “Attention” is used and divided into the “High”, “Medium” and “Low” values (see Fig. 4.2). This variable takes as input values the measures of face attention estimation considered by HP (Eq. 4.2). In Fig. 4.2 it is possible to see the labels for the variable “Attention”.

4.1.2 Fuzzy system for interest estimation

Once the three linguistic variables have been defined, the rule base that integrates them is explained in this section. The idea that governs the definition of the rule base is dominated by the value of the variable *Attention*. If *Attention* has a high value then the possibility of interest is also high depending on the distance and the angle of the person to the robot. If *Attention* is medium then the possibility of interest has to be decreased but, like in the former case, depending on the distance and the angle. Finally, if *Attention* is low, it means that the person is not looking at the area where the robot is located and the possibility of interest is defined as low or very low depending on the other variables. The rules for the case in which *Attention* is High are shown by Table 4.1. The other cases are expressed in a similar way using the appropriate rules. The output linguistic variable is $Interest$ and has the five possible values shown by Fig. 4.2(d).

Table 4.1: Rules in the case of high Attention.

IF			THEN
Attention	Distance	Angle	Interest
High	Low	Left	High
High	Low	Center	Very High
High	Low	Right	High
High	Medium	Left	Medium
High	Medium	Center	High
High	Medium	Right	Medium
High	High	Left	Low
High	High	Center	Medium
High	High	Right	Low

Finally in order to compute the value of possible interest, a fuzzy inference process is carried out using the minimum operator as implication operator. Then the output fuzzy sets are aggregated and the overall output is obtained. The overall output fuzzy set can be understood as a possibility distribution of the interest of the person in the $[0, 1]$ interval. Therefore values near to 1 mean a high level of interest and vice versa.

4.2 Recognizing typical interaction situations

After estimating the interest as described in the previous section, it is desirable that the robot centres its attention in the person that is more interested in interacting with it. This person could possible be willing to communicate with the robot in different ways.

The goal in this section is to compute whether a person, whose level of interest estimated before is high, is requesting attention from the robot. We are interested in the analysis of some typical interaction situations that can be integrated in a more complex system. The proposed situations are: i) the interaction demanding through the position or motions of the arms; ii) the shaking and nodding of the head to express assent or negation. These analysis are carried out using visual information and dealing with its underlying uncertainty and vagueness by means of FL.

After detecting the level of interest among those people located in the surroundings of the robot, the system detects if the person is static (not moving or moving very slowly). If so, the system analyses whether the person is standing (rising or extending) one or both arms as well as whether he/she is doing any movement with any of them.

As during an interaction people tend to ask and answer typical yes/no questions, a method employed to detect whether the ”interested” person is shaking or nodding his/her head has been developed. This feature can be employed in a more complex system, when the robot is able to, for instance, ask questions to the user.

4.2.1 Gestures Detection

One of the most common ways to request somebody attention using gestures is to raise or to shake one or both arms. Therefore efforts were focused in detecting whether a person, who might be interested in communicating, is doing this kind of gestures or not.

Using the information supplied by the SV system it is possible to know the position and distance at which many of the image pixels are located, and therefore to compute which objects are part of the foreground. To

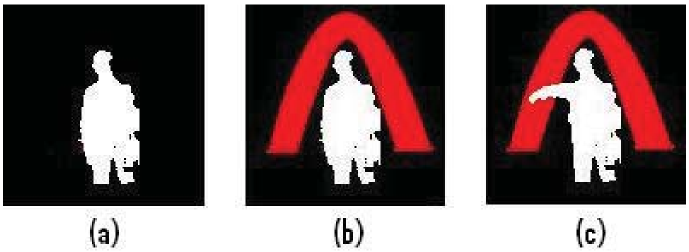
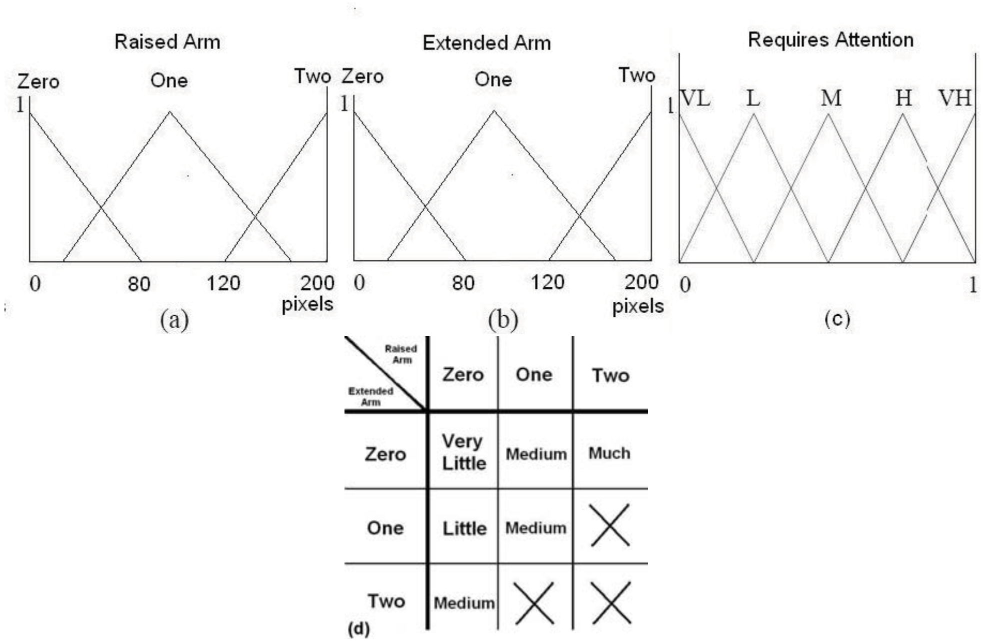


Figure 4.3: (a) Silhouette of a person. (b) Silhouette image marked with area for detecting raised arms (red). (c) Silhouette with red marked area and with a raised arm inside it.

achieve that goal an algorithm based on the “Distance of Mahalanobis” is used to separate the pixels that are part of the background from those who belong to the foreground. The “Distance of Mahalanobis” is based not only in the euclidean distance but also in the correlation between two variables. Afterwards, the objects and people belonging to the foreground are separated, using the information of the position of each person given by the tracker. A recursive algorithm called “Flood Fill” is then employed to build an image of the silhouette of that person. This algorithm computes whether the pixels surrounding the root pixel (the pixel assigned to be the centre of mass of the person) are within a specific “distance” and, therefore, also belong to that person. If so it continues to search for pixels in the surroundings of the new pixels that were previously classified as belonging to the person. When no more pixels satisfy this condition the image of the mask of that person is obtained where each pixel that belongs to the person has the information about its distance to the camera and pixels not belonging to the person are set to value 0. A more simple version of this image is the silhouette of the person or the binary image of the person as seen in Fig. 4.3(a). In this picture, pixels in white belong to the person while pixels in black do not belong to the person.

By doing this, it is possible to analyse if there are pixels around the person body that could be part of a raised arm. In the hereby presented system, all pixels that are not set to 0 in an elliptic region around the person (see Fig. 4.3(b)) are chosen. The inner border of the ellipse is just next to the person exterior border while the outer border of the ellipse is located more exteriorly in a way that any possible raised arm could fit inside the elliptic region. In Fig. 4.3(c) it is possible to see an example of the silhouette of a person whose arm is inside the elliptic region, indicating that the person is raising it or moving it. The algorithm only examines the upper half of



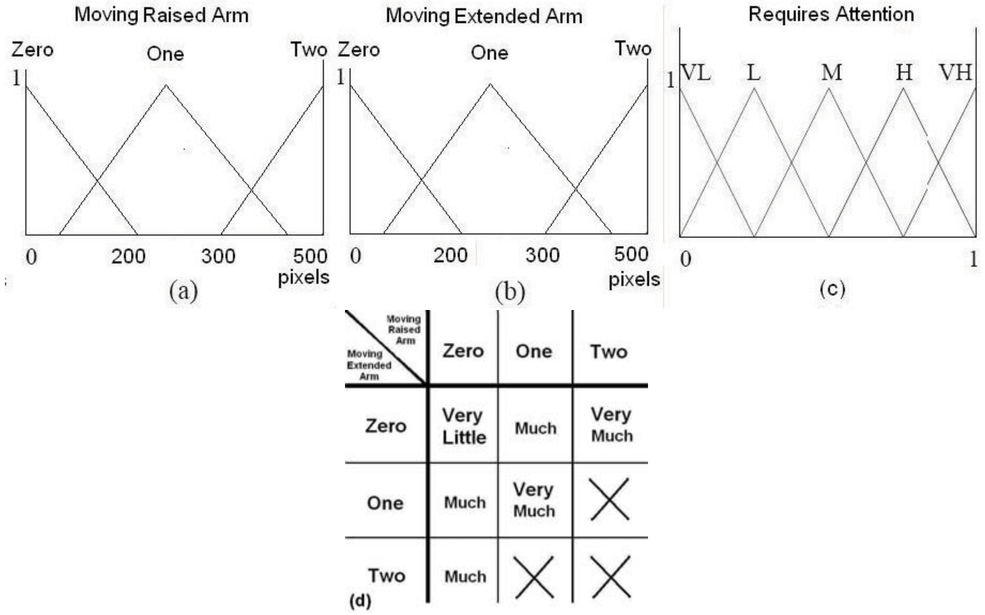
Raised Arm / Extended Arm	Zero	One	Two
Zero	Very Little	Medium	Much
One	Little	Medium	✗
Two	Medium	✗	✗

Figure 4.4: Fuzzy sets of the linguistic variables: (a) Raised Arm (b) Extended Arm (c) Requires Attention Position (d) Rule base of the FS

the ellipse because people arms can never be below the hips whenever a person is standing up. The linguistic variable *RaisedArm* is defined and it can take three values represented by the labels “Zero” “One” and “Two” (see Fig. 4.4) that represent the number of arms inside the region according to the number of pixels.

As it is also possible that the person is moving his/her arms forward in the region between the robot and the person, the system also analyses the distance between each of the person pixel to his/her mass centre. By doing so, it is possible to analyse the number of pixels that are not close to the mass centre and that could potentially be part of an extended arm. The linguistic variable *ExtendedArm* is defined in order to represent this situation. This variable can also take three values represented by the labels “Zero”, “One” and “Two” (see Fig.4.4) that represent the number of arms inside this region according to the number of pixels.

To analyse whether a person is moving an arm instead of only raising or extending it, the system analyses the number of pixels in the last frames building an image made of the pixels existing in the elliptic area (for a raised arm) or in front of the person (for an extended arm) in the last frames. If a person is moving one arm in that area, the number of pixels



Moving Extended Arm \ Moving Raised Arm	Zero	One	Two
Zero	Very Little	Much	Very Much
One	Much	Very Much	X
Two	Much	X	X

Figure 4.5: Fuzzy sets of the linguistic variables: (a) Moving Raised Arm (b) Moving Extended Arm (c) Requires Attention Movement (d) Rule base of the FS

in the last frames should be higher (two or three times more) than for one arm that is only raised or extended. Analogously, two linguistic variables $MovingRaisedArm$ and $MovingExtendedArm$ are defined. They can also take values "Zero", "One" and "Two" (see Fig. 4.5) according to the number of arms which is given by the number of pixels in both regions as previously described.

This information is given to two parallel FS that compute the level of attention demand of each person. The first one computes the level of attention demand using only the information about the number of raised and extended arms that are not moving (based only in position) while the second one does the same using the information about moving arms in both regions. The output from the first FS is called RAP (Requiring Attention Position) while the output from the second is called RAM (Requiring Attention Moving). These linguistic variables are represented in Fig.4.4 and Fig.4.5. The rule base for each one of the two FS is represented in Fig. 4.4(d) and Fig. 4.5(d). The fuzzy variable $RAfuzzy$ (Requesting Attention fuzzy) takes the maximum value of variables RAP and RAM:

$RAfuzzy = max(RAP, RAM)$

The defuzzyfied value of $RAfuzzy$ belongs to the $[0, 1]$ interval and represents the instantaneous level of attention demand. This value is weighted with past observation in a similar way than the shown in Section 4.1.1.

4.2.2 Shaking or nodding of the head

When communicating with people, head shaking or nodding is normally used to express agreement and disagreement with others. Similarly, during an interaction between a robot and a person, it might be important to detect if the person is agreeing or disagreeing, respect to some statement or situation during the interaction process. Speech recognition could be used to detect this kind of situation, but as people tend to shake and nod their head while they speak, it turns out to be an interesting feature to add by recognising this kind of gestures.

In order to detect this kind of situation, the system uses the area of the face given by the face detector. After obtaining the face region, it is applied a Sobel filter to extract the gradient of the face. Then it is possible to analyse the direction in which the face has moved from the previous to the current frame. For that purpose, we compare whether the region of the current face image has shifted to the surrounding pixels in the four main directions (up, down, left and right). To achieve this, the system computes the difference between the previous and current image gradients (see the following equation).

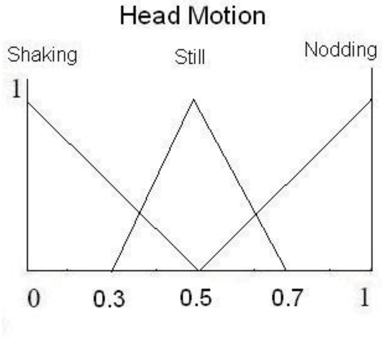


Figure 4.6: Labels for variable Head Motion.

$$SD = \sum_{i=0}^{n} \sum_{j=0}^{m} |p(i,j)_t - p(i,j)_{(t-1)}| \tag{4.3}$$

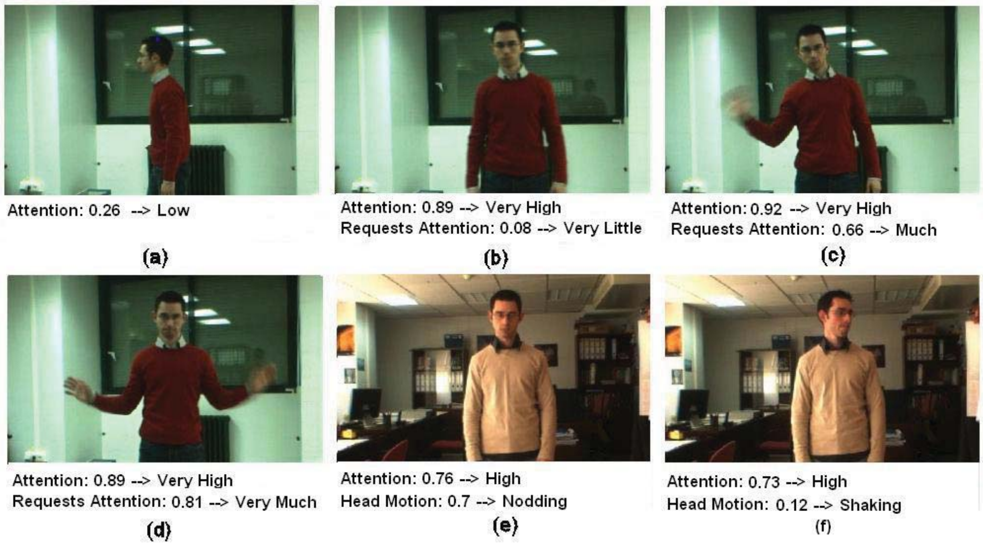

Attention: 0.26 --> Low
(a)
Attention: 0.89 --> Very High
Requests Attention: 0.08 --> Very Little
(b)
Attention: 0.92 --> Very High
Requests Attention: 0.66 --> Much
(c)
Attention: 0.89 --> Very High
Requests Attention: 0.81 --> Very Much
(d)
Attention: 0.76 --> High
Head Motion: 0.7 --> Nodding
(e)
Attention: 0.73 --> High
Head Motion: 0.12 --> Shaking
(f)

Figure 4.7: Examples from the first video: Low interest (a), Highly interested and not moving the arm (b),Highly interested and moving one hand (c), Highly interested and moving both hands (d), Nodding the head (e), Shaking the head (f)

where n and m are the width and height of the face image and p(i,j) the gray level of pixel i,j.

Equation 4.3 is used to compare the current image with the previous image shifted by a variable offset. After some experiments it was experimentally defined that comparing images shifted in the four main directions until a maximum offset value of 5 pixels was enough to detect head shaking and heading.

At the end, it is chosen the one that has the smallest value, indicating that it is the image most similar to the current one. Therefore it is possible to know what is the direction (if any) that the face has moved and how many pixels has it moved (the speed at which the person is moving his/her face).

For each frame the system estimates in real time the direction of the head in one of the five categories: up (U), if the direction that had the smallest error was the upper direction, in down (D) if it was the down direction, in left (L) if it was the left direction, in right (R) if it was the right direction and in static (S) if it was not none of the above. This output is translated into a numerical value by the definition of the variable $HMcurr_t$. The value of $HMcurr_t$ in the time t is

$$HMcurr_t = \begin{cases} 1 & \text{if output = “U” or “D”;} \\ 0.5 & \text{if output = “S”;} \\ 0 & \text{if output = “L” or “R”.} \end{cases} \tag{4.4}$$

However, $HMcurr_t$ is an instantaneous value that does not take into account past observations. In order to consider past observations, the variable $HM_{(t)}$ is defined as:

$$HM_{(t)} = \alpha HM_{(t-1)} + (1-\alpha)HMcurr_t \tag{4.5}$$

where the initial value for HM is the first value of $HMcurr_t$ when the person is detected and α is a weighting factor that ponders the influence of past observations.

To deal with the uncertainty and vagueness existing in this process a linguistic variable called “Head Motion” is used and split into “Shaking”, “Still” and “Nodding” (see Fig. 4.6). This variable takes as input values the measures of the head motion estimation considered by HM_t (Eq. 4.5) so that values near 0 mean shaking the head and values near 1 mean nodding the head. In future works this fuzzy variable can be used to facilitate the communication with the people.

4.3 Experimentation

Several experiments have been done to validate our approach. The results were very satisfactory in respect to interest estimation, attention demand and shaking and nodding estimation. In this section two of the carried out experiments are described. To perform the stereo process, images of size $320x240$ have been done as well as sub-pixel interpolation to enhance the precision in the stereo calculation. The operation frequency of the hereby presented system is about 30 Hz without considering the time required for stereo computation.

Regarding interest estimation, it has been checked that the interest degree assigned to each tracked person increases and decreases dynamically accordingly to the behavior of the person in relation to the robot, i.e., it depends on whether the person is looking at the robot, on the distance from the robot, and on whether it is in front of it.

Fig.4.7 shows the first experiment with one person. In Fig.4.7(a) it is possible to see that the person is not looking at the camera and his level

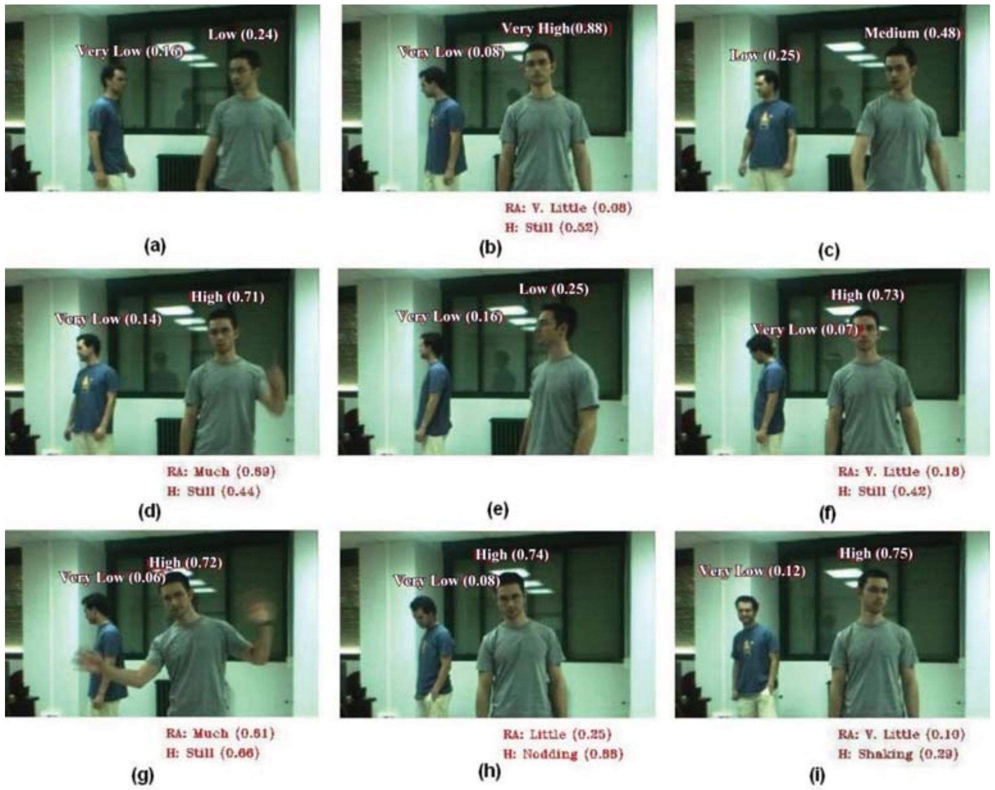


Figure 4.8: Examples from the second video: Both people with no interest (a), High interest from person on the right (b), Both people with little interest (c), High interest and demand of attention by the person on the right (d), Both people with no interest (e), High interest from person on the right (f), High demand of attention from person on the right (g), Person on the right nodding his head (h), Person on the right shaking his head (i)

of interest is low, while in the other frames his interest is higher. In frame Fig.4.7(b) the person shows interest but is not requesting attention because is not doing any movement with the arms. In frame Fig.4.7(c) the person is demanding attention because it is moving one of the arms. In frame Fig.4.7(d) the person is moving not only one arm but two, making the value of attention demand increase to higher values.

Concerning head shaking and nodding, it was possible to check that the system determined these movements most of the times. It was possible to check that head shaking was detected more accurately than nodding as the achieved results show (around 86% accuracy to head shaking while head nodding was about 83%). Fig. 4.7 shows in frames (e) and (f) examples of head nodding and head shaking and the output of the system for the first experiment.

With respect to the second experiment two people were used to test the system as seen in Fig. 4.8. In this experiment one of the people did not show interest to the robot (the person on the left) while the other one was changing his behavior over the time. The system never examines whether the person is demanding attention or making any movement with his head when his interest towards the robot is less than “High” as it is seen in frames (a), (c) and (e). In frame (a) it is possible to see that no person was showing interest to the robot. The same situation happens in frames (c) and (e) with different levels of attention for each of them. Between these frames the reader can see that the person on the right changed his behavior towards the robot. In frame (b) that person was showing interest to the robot but he was not demanding attention neither making any movement with his head, while in frame (d) the same person was requesting the robot attention by raising his left arm. In frame (f) and (g) similar situations to frames (b) and (d) are shown. Finally, in frames (h) and (i) there are examples of the person on the right nodding and shaking his head.

4.4 Summary and Final Remarks

In this work, a system for estimating the interest of the people in the surroundings of a robot and detecting motions related to attention demand and head nodding and shaking, is described. It uses SV, head pose estimation by SVM and FL. While a person is being tracked, the FS computes a level of possibility about the interest that this person has in interacting with the robot. This possibility value is based on the position of the person with respect to the robot, as well as on an estimation of the attention that the person pays to the robot. To examine the attention that a person pays to the robot the head pose of the person is analysed in real time. This analysis is performed by a view based approach using SVM. Thanks to SVM, the head pose can be detected achieving a great percentage of success that does not depend on the morphological features of the heads. The employed FS was also able to accurately detect demanding attention situation, by analysing the movement of the arms. The system also achieved a good result concerning the detection of head movements such as head nodding and shaking.

Chapter 5

Detecting Human Response Level

In the previous chapter an approach, in which the interest of each person towards the robot is calculated, was presented. In that approach, whenever the person was standing still and his/her level of interest was high, other kind of movements (arms movement and head shaking and nodding) were detected.

In the present chapter a slightly different approach is presented. In this work, the human response is computed without prioritising people that are closer than others respect the robot, people which are located more "centrally" than others regarding the robot or people expressly looking at it comparing to others that are not meeting this requirement. Now the idea is that the robot may play a role in certain social interaction activities in which it could interact with one or more people who are supposed to act in a natural way. This means that people might move freely in its surroundings, cross their paths and interact among themselves, as long as they respect the field of view and range of action of the camera. As in the previous Chapter, the robot is supposed to be immobile and observing the behaviour of each person. Distance plays an important role in choosing which variables are used but a person standing at 5 meters may present a much higher human response than someone which is closer to the robot. The algorithm also analyses other visual cues, such as the arms movement, while people are moving in its surroundings. In this work we try to improve the human behavior analysis by defining a human response measure based on visual cues which have different importance levels, depending on the distance of the person towards the robot.

5.1 Proposed Method

The proposed system is aimed to be used in indoor environments and it is prepared to work (depending mainly on the performance of the stereo sensor) in a range that might vary from d_{min} to d_{max} meters. With the used Stereo Vision (SV) sensor, $d_{min} = 0.5\ meters$ and $d_{max} = 5\ meters$ were defined. It is able to deal with at least 4 people at the same time (restrictions imposed by real time processing capabilities and the kind of camera used in this work). These operating variables could be increased in the future by simply using more powerful sensors and processors.

In order to analyse the response of those people to its interacting actions, the robot captures several features (input variables) which are described in Section 5.1.1. These features are the attention that each person is paying to him (face pose regarding the robot), occlusion (considering that people try to avoid obstacles when interacting with it), certain arms movements (the robot might ask people to interact by shaking their arms) and smile detection (it should provide a good feedback about each person satisfaction level according to those actions and messages transmitted by the robot). In this work, variables which could be detected by a SV system were prioritised.

In Fig. 5.1 it is possible to see how the visual perception of one person may change depending on the distance of the person towards the camera. Therefore, the variables used to compute the human response, based on visual cues, also depend on the distance at which each person is located regarding the robot. Then, Fuzzy Logic (FL) is the tool used to fuse all these information cues as it allows the handling of the uncertainty and vagueness that come along with the information provided by the sensors. It also allows to handle knowledge with straight forward rules defined in a human alike way and to easily fuse different output from different Fuzzy System(s) (FS). In Section 5.1.2 these FS are presented as well as an explanation of how the outputs of those FS are merged, according the distance of each person to the robot.

Finally, by computing the human response for certain number of frames, it is possible to analyse how a specific social-interaction activity proposed by the robot has an impact on a person. Also, by analysing the human response during the whole time of the interaction, it is possible to compute the success of the whole activity. Both measures provide important information which might lead to the improvement of the proposed activity. These measures are explained in Section 5.1.3 more in detail.

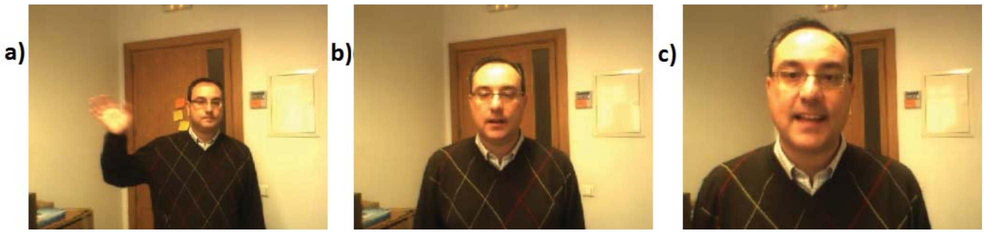


Figure 5.1: Different visual perceptions of a person depending on the distance: person located at about about 3 meters (a), at about 1.5 meters (b) and at about 1 meter (c).

5.1.1 Input Variables

In this section, the variables used to compute human response values are described. Firstly, a description of the variable related to the detection of a frontal face is given and then we pass over to the variable that represents occlusion. Variable that represents arms movement is explained next and the variable which represents smile detection is lastly detailed at the end of this section.

Frontal Face

In order to detect whether someone is looking at the camera, a frontal face detector [Sou10] is used. This face detector is based on the Viola and Jones' method [VJ01]. Their detector uses Haar based features. To select the specific Haar features to use, and to set threshold levels, Viola and Jones use a machine-learning method called AdaBoost. In Fig. 5.2 both examples of a frontal and non frontal face are shown.

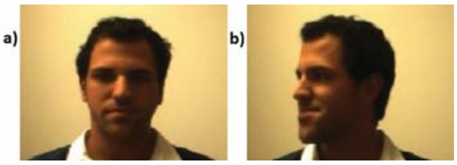


Figure 5.2: (a) Example of a frontal face. (b) Example of a non frontal face.

With the goal of determining that someone is looking at the camera, we consider that this situation happens when the system detects a frontal face during some instants of time. We have experimentally defined that this information should be obtained by analysing the current frame and the 9 previous frames totalising 10 frames. Considering this number of frames an average value, $AverageFrontalFace(P)$, is obtained to be used as input to FS.

More specifically, the variable $FrontalFace(P)$ is defined as:

$$FrontalFace(P)_t = \begin{cases} 1 & \text{if } detectedFace(x_p, y_p)_t = true \\ 0 & \text{if } detectedFace(x_p, y_p)_t = false \end{cases} \tag{5.1}$$

where P represents a detected person and (x_p, y_p) his or her face central position in the reference image.

Finally variable $AverageFrontalFace(P)$ is defined as:

$$AverageFrontalFace(P)_k = \frac{\sum_{i=k-9}^{k} FrontalFace(P)_i}{k}, \text{ with } k = 10 \tag{5.2}$$

This variable and the FS that deals with this information help filtering instantaneous false positives or negatives of the face detector, making this information more robust to errors.

Occlusion management

In Chapter 3, Section 3.2.1, the concept of "projection of a person" was presented. Basically, it can be viewed as the 2D region that both face and torso of a standing up person, with average size, would occupy on the reference and on the distance image, if his or her face were approximately on the same 3D location of the detected face. Therefore, one region projected in the reference camera image, made of two elliptical regions (corresponding to head and torso), and denoted by RP_{ri}, is defined. Similarly, two elliptical regions (head and torso) projected in the distance image, denoted by RP_{di}, are also defined. The size of these ellipses takes into account the distance of the person to the camera. Fig. 3.3 shows those regions for two different people both in the reference image ($RP_{ri}1$ and $RP_{ri}2$) and in the distance image ($RP_{di}1$ and $RP_{di}2$).

In this work, the concept of occlusion map is also used according to the definition given in Section 3.2.1. It is thus possible to compute the number of occluded pixels regarding each person.

Concerning the human response detection algorithm, it is assumed that someone who is less occluded than another person, should have an higher level of human response. Variable $AmountNonOccluded(P)$ (with value between 0 and 1) was defined according to:

$$AmountNonOccluded(P)_t = \frac{pixelsNotOccluded(P)_t}{totalNumberPixels(P)_t} \tag{5.3}$$

where variable $pixelsNotOccluded(P)$ represents the number of pixels of the "projection of the person P" which are not occluded by another person and variable $totalNumberPixels(P)$ represents the total number of pixels of the "projection of the person P".

Arms Movement Computation

In order to compute arms movement, a previous algorithm described in Chapter 4 is employed. The first step is to compute the mask image of each person, concept which is also used for filling up the occlusion map, where each pixel that belongs to the person has the information about its distance to the camera and those pixels that do not belong to the person are set to 0. In Fig.5.3(b) and Fig.5.4(b), the regions in grey represent the mask associated to a person.

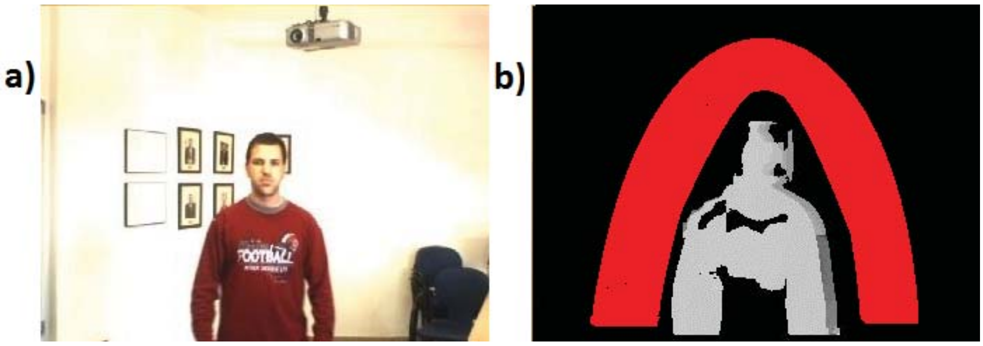


Figure 5.3: (a) Reference image with one person not shaking any arm. (b) Distance image of the reference image with a red ellipse determining the zone where the system looks for extended arms.

After determining the mask associated to a person, it is possible to search or to infer about the situation of having pixels around his or her torso that could potentially be part of a raised arm. In this approach the first step is to look for pixels that contain distance information in an elliptic region that surrounds the person body (see red ellipse in Fig. 5.3(b) and in Fig. 5.4(b)). This region, where each arm of the person may be located, whenever his or her arm is raised, is defined as seen in Fig.5.4(b). Different fuzzy variables were defined in order to represent the number of arms (static or moving) inside that area as described in Section 4.2.1.

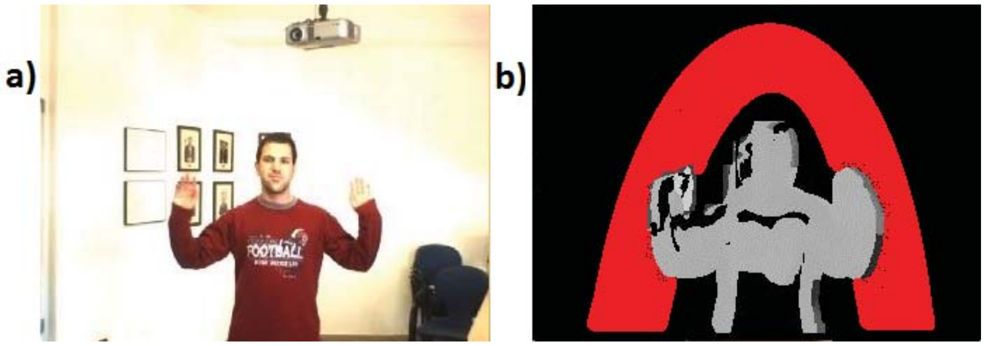


Figure 5.4: (a) Reference image with one person shaking both arms. (b) Distance image of the reference image with a red ellipse determining the zone where the system looks for extended arms.

Another circumstance to consider is that it is possible that the tracked person might be shaking his or her arms in an "extended" way and in the direction of the camera. So, the detection has to be done in a zone close to his or her torso, between the person and the camera. By analysing the distance at which each pixel of the person mask is located, and by comparing this set of distances to his or her mass centre, it is possible to infer if there is a significant region of his or her body that is slightly forward. The possibility that the extended arms are moving is also taken into account. In case it exists, this region is potentially an arm that is located in the space between the person and the camera. Once again, different fuzzy variables are defined to represent the number of arms (static or moving) inside that region as described in Section 4.2.1.

To compute the final value of the arms movement produced by the situation of raising/extending, a static/moving arm, two parallel FS are used as described in Section 4.2.1 of Chapter 4. Arms Information or $AI(P)$ represents the output of those FS used to compute the level of arms movement. It is a value between 0 and 1 which is weighted, taking into consideration the previous information of each person arms movement, according to the next equation:

$$AI(P)_t = \delta AI(P)_{(t-1)} + (1-\delta)AI(P)_t \tag{5.4}$$

where δ is a threshold value experimentally determined to set the weight of past or current measures.

Smile Detection

Here, machine learning is used to classify the different situations. As described in Section 2.3.2, Support Vector Machines (SVM) is a technique to analyse and classify data according to given patterns. For each input, SVM predicts which of different classes forms the input. In this approach, one of the cues that are used in order to analyse human behaviour, is the smile. In this case, it is assumed that a person might be in two situations: smiling or not. This kind of binary classification makes it adequate for using a SVM based classifier. To use SVM, a database of examples should be firstly set up containing enough examples for the SVM classifier algorithm to learn and build a model capable of assigning new examples to each of the categories.

In this approach, face images of people either smiling or not smiling are classified in two different categories (*Smiling* or *NotSmiling*). Some examples of these images can be found in Fig.5.5.

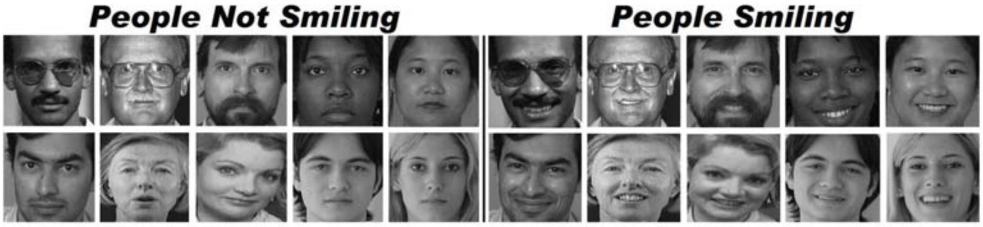


Figure 5.5: On the left part of the image there are 10 examples of non-smiling faces and on the right part of the image there are 10 examples of smiling faces.

As images often contain a lot of information which is redundant and therefore less important for the classification system, a first step is advisable to extract the most important information of those images. In order to reduce the data dimensionality, Principal Component Analysis (PCA) has been applied (see Section 2.3.1 for more information about PCA). Tests with different number of components were performed and the number of 100 components was considered to be very satisfactory to achieve a good trade-off between classification accuracy and computing time.

The libsvm library [CL11] was chosen for classifying using SVM. A database of roughly 2000 frontal face images (40x48 pixels) was built where one half of these images were marked as belonging to the *smiling* class and the other half was marked as belonging to the *nonsmiling* class. The face images used have been provided by different public available databases namely several from the Computer Vision Laboratory, University of Ljubljana, Slovenia, [Pee03, SPB$^+$03] from the FERET database of facial images collected under the FERET program and sponsored by the DOD Counterdrug Technology Development Program Office [PWHR98, PMRR00], and from the ORL Database of Faces [SH94]. Some other image faces from people belonging to the University of Granada, Spain were added to the previously mentioned databases. In order to validate the proposal, cross validation was applied. The accuracy obtained with this method was of 89.2%. Before deciding to use this methodology, others were tested namely a classification based only on the mouth region of the face (after applying a mouth detector) which was abandoned due to the fact that sometimes mouth region images that look like as "not smiling" belong to people which are smiling and vice-versa. Other feature reduction techniques as the Discrete Cosine Transform were tested on the whole database of face images but this method did not prove to an improvement of the accuracy regarding the PCA used method.

After the training step, face images extracted from tracked people during the execution of the algorithm were classified in one of those two mentioned categories: *Smiling* or *NotSmiling*. To transform this information in an usable value, the variable *Smile*(*P*) was defined as:

$$Smile(P)_t = \begin{cases} 1 & \text{if } detectedSmile(P)_t = true \\ 0 & \text{if } detectedSmile(P)_t = false \end{cases} \tag{5.5}$$

Finally variable *AverageSmile*(*P*) was defined as:

$$AverageSmile(P)_k = \frac{\sum_{i=k-9}^{k} Smile(P)_i}{k}, \text{ with } k = 10 \tag{5.6}$$

This variable and the FS that deals with this information help filtering instantaneous false positives or negatives of the smile detector making this information more robust to errors.

5.1.2 Fuzzy Systems for Processing Information

In this section, the hierarchical structure of the built FS and each of these FS, used to compute the final value of human response, are described. On the definition of these FS, the physical impossibility to detect the same features at different distances was taken into consideration. In Fig.5.1 there are 3 screenshots of the same person at different distances from the robot. For instance, when someone is closer than a certain distance respect to the stereo camera (which is considered as the eyes of a robot), not all the required features can be registered by the camera. As an example of this situation, if a person is moving his or her arms very closely to the camera, the field of view of the camera does not allow to detect this movement. Also, when someone is located farther from the camera, the precision of the used camera does not allow to detect whether that person is smiling or not.

Two main FS were defined according to features which can be detected at different distances. The range of distances at which each feature is still detectable was experimentally computed. A third FS is used to fuse the outputs of these FS, according to the distance at which the detected person is located. A scheme of the FS structure can be seen in Fig. 5.6.

Now, the three FS are going to be explained.

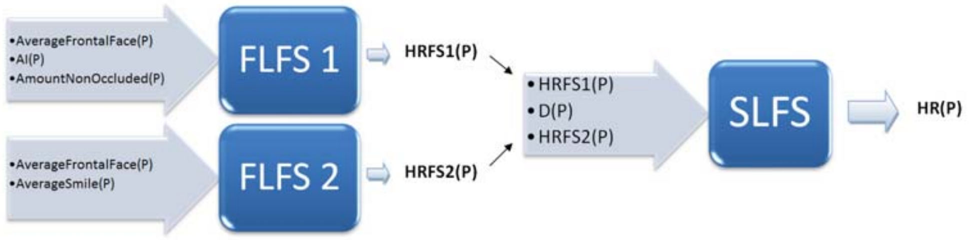


Figure 5.6: Hierarchical Structure of Employed Fuzzy Systems

First Layer Fuzzy System 1: Features Detected at Farther Distances

This FS is used within a specific range of distances between β and d_{max} (maximum distance value defined for this camera). This range is what it will be called from now on "farther distances". In this work, d_{max} is 5 meters and β was estimated as 2.0 meters.

The first step is to define the three linguistic variables that are used to fuzzify the information coming from the three different sources of information detected at farther distances, that is, $AverageFrontalFace(P)$, $AI(P)$ and $AmountNonOccluded(P)$. The three linguistic variables are called $FrontalFaceFuzzyFar$, $ArmsInformationFuzzy$ and $OcclusionInfoFuzzy$ and they take its input values from variables $AverageFrontalFace(P)$, $AI(P)$ and $AmountNonOccluded(P)$ respectively. Smile is impossible to be detected at this distance with the used camera. Each of these variables is represented in Fig.5.7(a), (b) and (c) where it is possible to see the different labels defined for each of these variables.

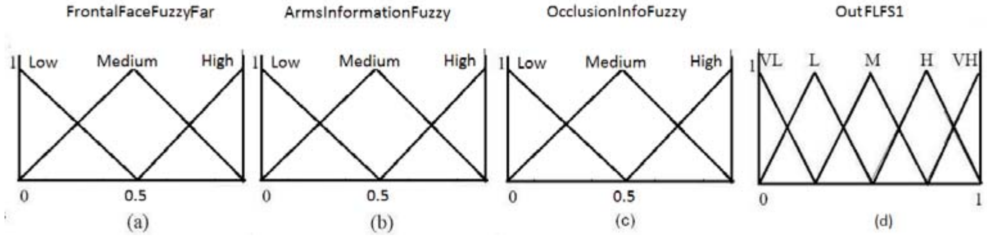


Figure 5.7: Fuzzy sets of the linguistic variables: (a) FrontalFaceFuzzyFar (b) ArmsInformationFuzzy (c) OcclusionInfoFuzzy (d) OutFLFS1.

The next step is to define the output variable, $OutFLFS1$, which is represented in Fig.5.7(d). This output variable has 5 different labels which are also represented in Fig.5.7(d).

Another key step is to define the rule base for this FS. The rules used in the defined FS are available on Table 5.1 and were experimentally defined. The maximum value of the output of this FS is 1 whenever the person is interacting, by shaking his or her arms and looking at the camera without being occluded. The output decreases if the person is not complying with one or several of these conditions reaching 0, the minimum value, whenever the person is neither looking at the camera nor moving his or her arms while most of his or her torso and face are occluded.

Finally, to compute the value of human response in this context, $FLFS1$ uses the Mamdani inference method. Then the output fuzzy sets are aggregated and the overall output is obtained. The overall output fuzzy set is defuzzified and can be seen as a measure of the human response of the tracked person P in the [0,1] interval, at farther distances. $HRFS1(P)$ is denoted as the defuzzified output of $FLFS1$. Therefore a value of $HRFS1(P)$ close to 1 means a higher level of human response than a value closer to 0.

First Layer Fuzzy System 2: Features Detected at Close Distances

Firstly, the range for what will be called “closer distances” is defined. Thus, this FS is used within a specific range of distances between d_{min} (minimum distance value defined for this camera) and α. In this case, d_{min} was defined as 0.5 meters and α was defined as 1.0 meter.

For those people located at a distance lower than α meters and higher than d_{min} meters, arms movement and occlusion make no sense to be considered. So, it is necessary to define another FS which takes into consideration those features which can be detected at lower distances. These features

Table 5.1: FLFS 1 Rules.

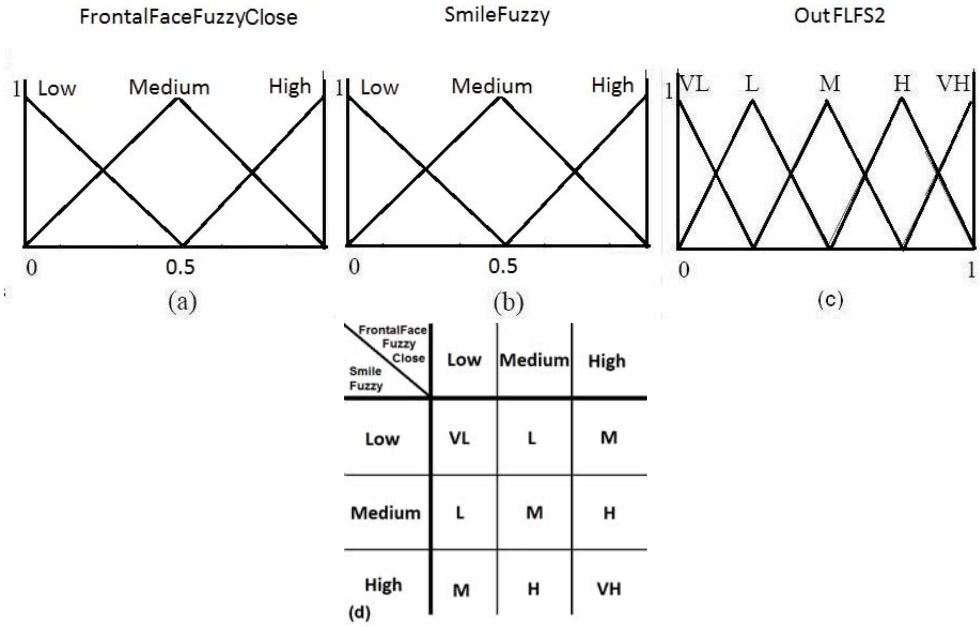
IF			THEN
FrontalFaceFuzzyFar	ArmsInformationFuzzy	OcclusionInfoFuzzy	OutFLFS1
High	High	High	H
High	High	Medium	VH
High	High	Low	VH
High	Medium	High	M
High	Medium	Medium	M
High	Medium	Low	H
High	Low	High	L
High	Low	Medium	L
High	Low	Low	M
Medium	High	High	L
Medium	High	Medium	M
Medium	High	Low	H
Medium	Medium	High	L
Medium	Medium	Medium	L
Medium	Medium	Low	M
Medium	Low	High	VL
Medium	Low	Medium	VL
Medium	Low	Low	L
Low	High	High	L
Low	High	Medium	L
Low	High	Low	M
Low	Medium	High	VL
Low	Medium	Medium	VL
Low	Medium	Low	L
Low	Low	High	VL
Low	Low	Medium	VL
Low	Low	Low	VL

are: “frontal face detection” and “smile detection” which are computed by the variables $AverageFrontalFace(P)$ and $AverageSmile(P)$ respectively.

In order to do so, the first step is to define those 2 linguistic variables that are used to fuzzify both features. They are called $FrontalFaceFuzzyClose$ and $SmileFuzzy$ which take its input value from variables $AverageFrontalFace(P)$ and $AverageSmile(P)$ respectively. Each of these variables is represented in Fig.5.8(a) and Fig.5.8(b) where it is possible to see the different labels defined for each of these variables. The next step is to define the output variable, named $OutFLFS2$. This variable is represented in Fig.5.8(c) with its 5 different labels.

As for the FS used for farther distances, the rule base for this FS was defined. The rules used on the defined FS are available in Fig.5.8(d) and were also experimentally determined.

The same inference process is used as for the FS used for farther distances. The output is also defuzzified and corresponds to a value in the [0,1] interval. $HRFS2(P)$ is denoted as the defuzzified output of $FLFS2$. Therefore a value of $HRFS2(P)$ close to 1 means a higher level of human response at close distances than a value closer to 0. The maximum value



FrontalFace Fuzzy Close / Smile Fuzzy	Low	Medium	High
Low	VL	L	M
Medium	L	M	H
High	M	H	VH

Figure 5.8: Fuzzy sets of the linguistic variables: (a) FrontalFaceFuzzyClose (b) SmileFuzzy (c) OutFLFS2 (d) Rule base of the fuzzy system $FLFS2$

of the output of this FS is 1 whenever the person P is smiling and looking at the camera. The output decreases if the person is not complying with one of these conditions reaching its minimum of 0 when the person is not looking at the camera and not smiling.

Second Layer System: Fusing Information According to Distance

Another consideration is that for distances up to α meters, features like frontal face detection and smile detection worked with little error while the goal of detecting the movements of the tracked person arms was almost impossible to achieve. Also, for distances farther than β meters, smile detection is not possible with the used camera. Features like arm movement detection and occlusion detection are correctly detected and features like frontal face detection gradually are more difficult to detect regarding closer distances. Finally, for those distances between α and β meters, all features could potentially bring valuable information to the system, although this information still depended on specific situations (at a α+0.1 meters distance, the smile detector detects the smile accurately but at $\beta - 0.1$ meters the chance that an erroneous smile detection occurs is higher). These thresholds might be adjusted on future works depending on the characteristics of the hardware such as the resolution or the field of view of the camera.

A way to deal with this situation is to create another FS which handles the outputs of each of the previously mentioned FS together with the distance information of the person who is being analysed and outputs the final human response level. Thus, $SLFS$ has tree input variables. Two input variables are crisp values ($HRFS1$ and $HRFS2$) and the third input is the linguistic variable $DistancePerson$ which is shown by Fig.5.9. $DistancePerson$ is used to fuzzify the value of the person distance $D(P)$ towards the camera, which can be computed with the stereo system. The final output variable was defined and called $HumanResponse$ or $HR(P)$ where P is one of the person being tracked.

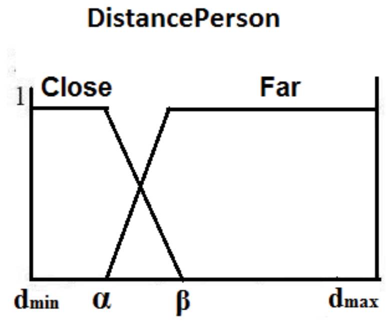


Figure 5.9: Linguistic variable DistancePerson

$SLFS$ has two fuzzy rules:

If $DistancePerson = Far$ then $HR(P) = HRFS1$
If $DistancePerson = Close$ then $HR(P) = HRFS2$

Therefore the final value HR is computed by:

$$HR(P)_t = \mu_{(Close)}D(P)_t * HRFS2(P)_t + \mu_{(Far)}D(P)_t * HRFS1(P)_t \quad (5.7)$$

where $\mu_{(Close)}$ is the membership function of the $Close$ label, $\mu_{(Far)}$ is the membership function of the Far label and $D(P)$ is the distance of the person P towards the camera.

$HR(P)$ is a value in the interval [0,1] and represents the final human response computed by the FS for a person P. A value of HR near to 1 means a good human response, and a value near to 0 means a poor human response.

5.1.3 Instantaneous and accumulated results

From the mathematical expression presented in Eq. 5.7, an instantaneous human response measure is obtained, for each person, at each frame. This value is able to indicate if that person is responding to the actions purposed by the robot in the frame t.

By taking into consideration all the values of instantaneous human response for each person, during the whole activity, one is able to see at which moments people responded better to those actions performed by the robot. If all instantaneous human response values are summed up for each person, a global human response ($GHR(P)$) measure may be obtained:

$$GHR(P) = \sum_{t=1}^{N} HR(P)_t \tag{5.8}$$

where N is the final instant of the sequence for which the user wants to measure the global human response.

This measure is able to indicate which of those users showed a better human response for a specific period of time or for the whole activity. It is then possible to infer about the behavior of users in a specific action or activity.

In a similar way, by calculating the average value of $HR(P)$ as

$$\overline{HR(P)} = GHR(P)/N \tag{5.9}$$

it is possible to compute, for instance, which of the people participating in the experiments was collaborating the most with the robot.

By taking into consideration instantaneous, global and average values, one might analyse which aspects of the robot proposed activity might be improved or adapted. For instance, the robot may adapt its behavior whenever the feedback given by the users of the system is not satisfactory.

5.2 Experimental Results

In this section, the experiments carried out in order to validate the presented proposal, are presented. The achieved operation frequency of the system depends on different factors with the most important one being the number of people which are tracked at a time. Each tracked person uses an independent tracker (see 3), so processing time increases with the number of people. Considering the hardware and camera used, the system could be employed in real time activities for up to 4 people tracked at the same time.

In order to test the system, different colour-with-depth sequences were used. The data was recorded using the employed stereo camera. These videos were recorded in different environments such as different rooms and luminance conditions. The goal was to provide diversity with different background scenarios and conditions of use. Several people participated in the videos, moving without restrictions, with the freedom to interact among them and considering the robot as another person. The only request was that they thought about using those visual cues that the system is able to recognise (although these cues are part of those gestures that people normally use on their daily routines) when responding to the actions by the robot. The aim was to check whether the algorithm was able to correctly measure the human response on different situations.

Experiments were divided in two parts. The first one takes into consideration the situation of one person moving between the three different distance zones. The second one shows the functioning of the system when there are several people moving and interacting among them. Sections 5.2.1 and 5.2.2 show the results for the first experimentation while Section 5.2.3 shows the results for the second one.

5.2.1 Measuring the Human Response for One Person

In this section, frames taken from several videos where a person is standing at different distances are presented. Images of those situations are shown in Fig.5.10, Fig.5.11 and Fig.5.12. Below every image, the instantaneous human response value (between 0 and 1), corresponding to that frame can be observed. Also, there are different zones marked by rectangles of different colours. These rectangles correspond to the detected face for the person being tracked. Blue rectangles correspond to a face which is considered to be in a “smiling” situation and green rectangles correspond to a face which is considered “not smiling”.

Person standing at Farther Distances from the robot

Let us start by analysing the situation of one person at farther distances (Fig.5.10). In this case the person was standing approximately at 3.5 meters from the robot. As there is only one person, there is no occluding situation. Thus, $OcclusionInfoFuzzy$ value is always low. It is visible that the system always assigns the correct face rectangle to the tracked person. Also, there is no frame in which the system is able to detect the person smile, even on frame Fig.5.10(e). In Fig.5.10 it is also possible to see that $HR(P)_t$ is within an expected range for each of the shown situation. When the person is not looking at the robot and is not moving any arm (Fig.5.10(a)), his instantaneous human response is very low and when the person is staring at the robot and shaking his arms (Fig.5.10(f)) his instantaneous human response value is very high.

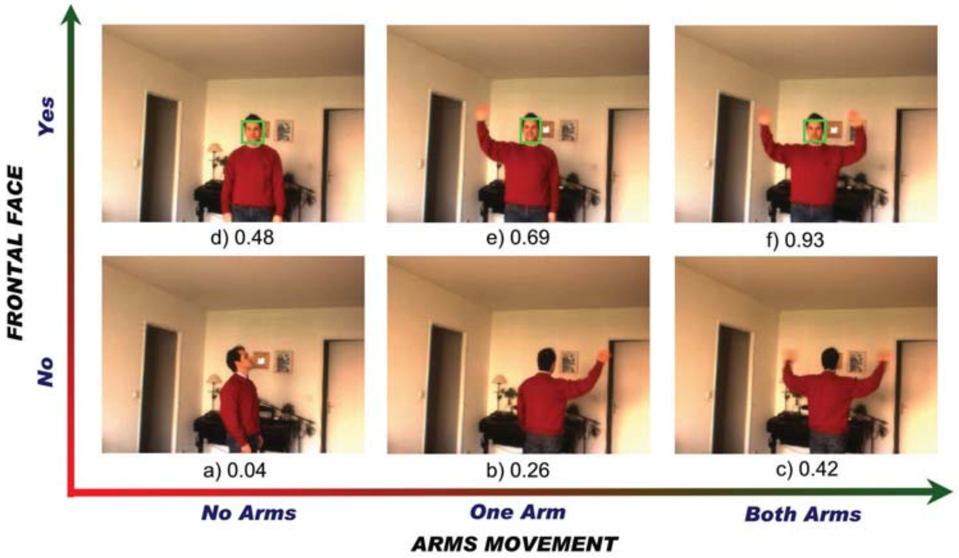


Figure 5.10: Instantaneous human response according frontal face and arms movement for a person located at farther distances (3.5 meters). Below each image the value of $HR(P)_t$ is indicated.

Person standing close to the robot

In this case (Fig.5.11), there is one person standing at approximately 0.73 meters from the robot. In this situation, as it is easy to observe, possible shaking arms would fall out of the screen image, even when the person is well centred in the camera image. In this situation, only frontal face and smile possibility are taken into consideration to detect the person human response. It is also observable that $HR(P)_t$ is according to the situation (i.e., for a person not looking at the robot) the human response value is close to 0 (frame Fig.5.11(c)), for a person looking at the robot and smiling during some frames, this value is practically 1 (see Fig.5.11(a)) and finally for a person looking at the robot but not smiling the value of $HR(P)_t$ is around 0.5 (Fig.5.11(b)).

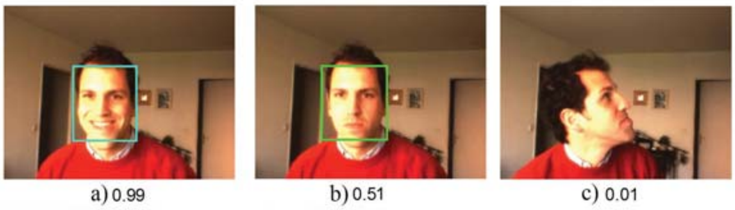


Figure 5.11: Instantaneous human response according frontal face and smile for a person located at close distances (0.73 meters). Below each image, the value of $HR(P)_t$ is indicated.

Person standing between Farther and Close Distances

Finally, examples of one person standing between Farther and Close Distances are shown. This corresponds to distances between α and β meters away from the robot as seen in Fig.5.12, where both FS (for faraway and close situations) are used. In this case, both FS are used to compute $HR(P)_t$ but its weight depends on whether the person is either closer to the farther area or to the closer area. In Fig.5.12 it is possible to see that when a person, with frontal face, is smiling and shaking his arms (Fig.5.12(d)), the human response value is almost 1 and when the person is not smiling neither shaking any of his arms (but still looking at the robot), his human response value decreases to almost 0.5 (Fig.5.12(a)). When the person either shakes his arms (Fig.5.12(b)) or smiles (Fig.5.12(c)), the human response value is approximately the same, as the person is approximately located halfway between α and β.

5.2.2 Transition among different situations

In this section, values of some of the variables used in the system, during a whole video of roughly 1500 frames, are shown. Basically, there is one person which is initially located 0.7 meters away from the robot and then moving backwards until a distance of 3.6 meters. During the video, the person walks and stops now and then for some seconds. The person is asked to use the previously presented interaction cues taken into account for computing human response. The goal is to observe the different FS output values and also how the variable $HR(P)_t$ depends on input variables such as $AverageFrontalFace$, AI and $AverageSmile$ as well as on the distance of the person towards the robot. In this experiment $OcclusionInfoFuzzy$ is always Low as there is only one person.

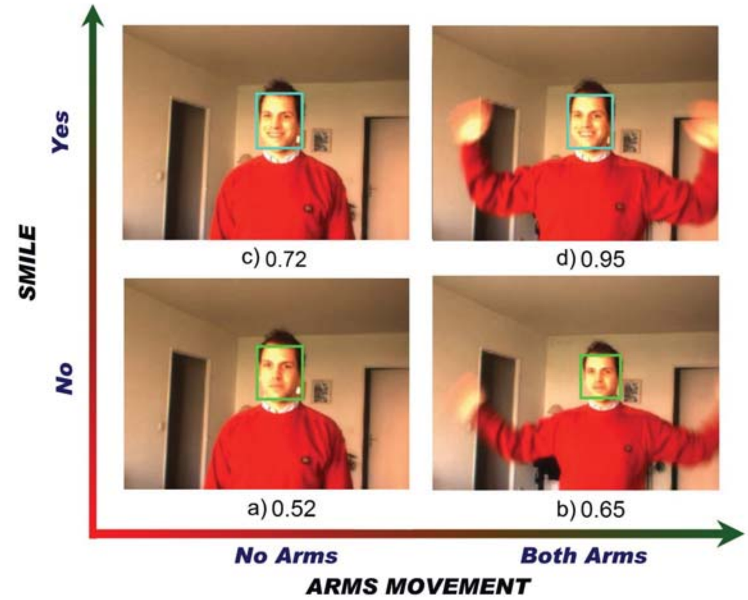


Figure 5.12: Instantaneous human response according arms movement and smile for a person with a detected frontal face who is located between Farther and Close Distances (between α and β meters). Below each image, the value of $HR(P)_t$ is indicated.

As defined in Equation 5.7, $HR(P)$ is equal to $HRFS2(P)$ whenever $D(P)$ is between d_{min} and α. Fig. 5.13 shows this situation and it is also possible to see how $HRFS2(P)$ depends on the input variables.

If $D(P)$ is between α and β, $HR(P)$ depends on both $HRFS2(P)$ and $HRFS1(P)$. In this case, the weight of each variable also depends on the distance of the person $D(P)$. Fig. 5.14 shows this situation and how $HR(P)$ depends on the input variables.

Finally, $HR(P)$ is equal to $HRFS1(P)$ whenever $D(P)$ is between β and d_{max}. Fig. 5.15 shows this situation and how the variable $HRFS1(P)$ depends on the input variables.

On the left part of Fig.5.13 it is possible to see that, when the variable $AverageFrontalFace$ is equal to 1, $HRFS2(P)$ mainly depends on the value of variable $AverageSmile$. When both variables are equal to 1, $HRFS2(P)$ is also equal to 1. On the right part of the graph one can see that $AverageSmile$ is never superior to $AverageFrontalFace$ as expected since it is assumed that a smile cannot be detected without a detected frontal face. When both input outputs are close to 0, $HRFS2(P)$ is also close to 0, as expected.

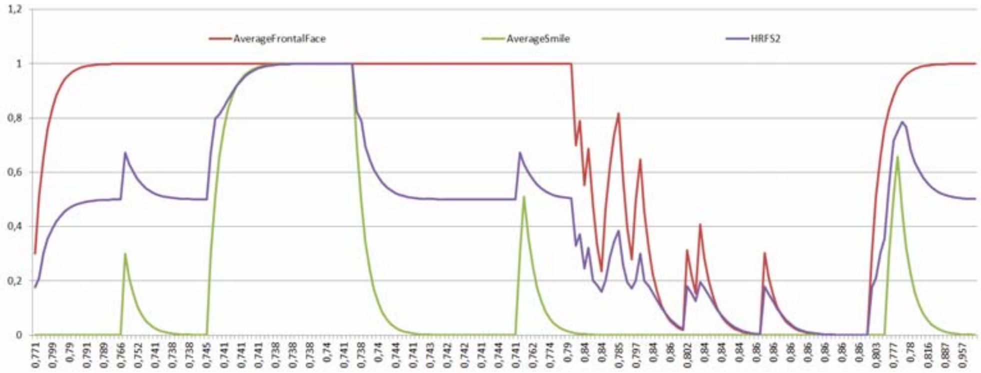


Figure 5.13: Graph showing $HRFS2(P)$ depending on input variables $AverageFrontalFace$ and $AverageSmile$ values (Y axis). The X axis represents the distance to the robot (always below α meters for this case which represents the close situation).

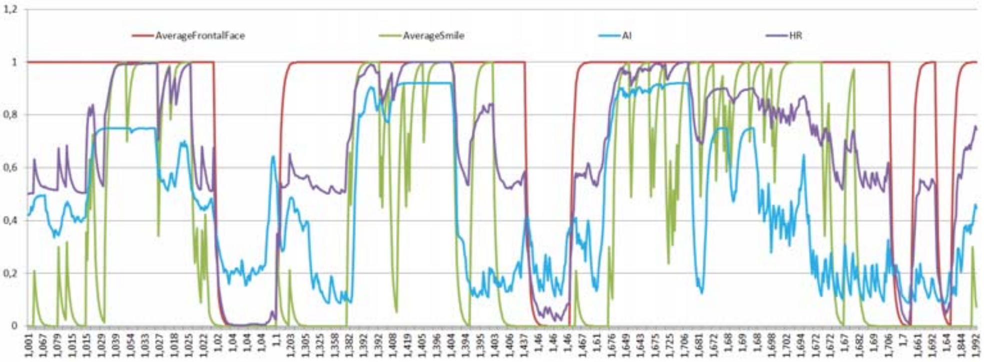


Figure 5.14: Graph showing $HR(P)$ depending on input variables $AverageFrontalFace$, AI and $AverageSmile$ values (Y axis). The X axis represents the distance to the robot (always between α and β meters for this case which represents the intermediate situation).

In Fig.5.14 it is possible to see that $HR(P)$ gradually starts to depend more on the variable AI than rather on variable $AverageSmile$ as the distances increases. This behavior can be seen, for instance, between 1.02 and 1.1 meters as $HR(P)$ practically reaches 0 because $AverageSmile$ and $AverageFrontalFace$ are 0, although AI is not 0. If we compare it to the situation at around 1.46 meters where $AverageSmile$ and $AverageFrontalFace$ are also 0, it is visible now that $HR(P)$ is slightly higher for roughly the same value of AI. This happens as the weight of AI is higher at 1.46 meters than at 1.05 meters. Another aspect to remark is the high instability of the smile detector respect to the close zone. This factor is due to a gradual lost of precision of the smile detector (isolated false positives or negatives). That is the main reason why the smile detector is not taken into account for positions above β meters and why its weight decreases when a person moves towards a distance close to the β threshold.

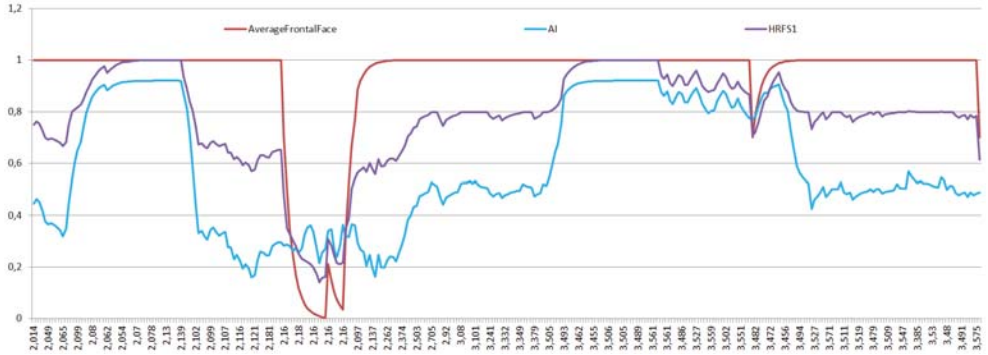


Figure 5.15: Graph showing $HRFS1(P)$ depending on input variables $AverageFrontalFace$ and AI values (Y axis). The X axis represents the distance to the robot (always higher than β meters for this case which represents the far situation).

In Fig.5.15 $HRFS1(P)$ only depends on $AverageFrontalFace$ and AI as there is no occlusion. As $AverageFrontalFace$ is mostly constant, it is easily observable that when AI increases or decreases, $HRFS1(P)$ also increases or decreases in a very similar way. When $AverageFrontalFace$ suddenly drops or rises, $HRFS1(P)$ behaves analogously.

Although some sporadic errors in the detection of the defined features can occur, this kind of graphs makes it possible to observe that the FS correctly outputs the expected human response values on most of the frames. The result is a smooth and natural recognition of the human response during almost the entire video, where those previously mentioned sensor errors tend to have less importance because of FL.

Concerning the processing time, it mainly depends on the used detection and tracking algorithm. An execution time of 20 and 30 ms for one tracking cycle per person was measured. Despite the high amount of data involved, we observed that the proposed algorithm is prepared to be used in real time environments and so allowing a natural interaction among all the people that participated in the experiments.

Table 5.2: Values of HR and D for P1 and P2

Frame	HR(P1)	HR(P2)	D(P1)	D(P2)
a	0	0.38	1.7	4.3
b	0	0.39	2.2	4.1
c	0	0.45	4.7	1.5
d	0.18	0.51	4.7	1.5
e	0.81	0.51	4.7	1.5
f	0.41	0.79	4.7	1.5
g	0.51	0.04	1.8	3.9
h	0.89	0	1.2	4
i	0.97	0	0.7	4
j	0.59	0	0.7	4
k	0.98	0	0.8	4
l	0	0	0.9	4.1

5.2.3 Multiple people

In this experiment, two people (denoted as $P1$ and $P2$) were asked to freely move in front of the robot, and to randomly choose between interacting among themselves or with the robot (using the previously defined gestures and actions). In Fig.5.16 there are some screenshots of one of the videos, and in Table 5.2 the corresponding value of HR and D for each person $P1$ and $P2$.

If the reader takes a deeper look into some of these situation, he can observe that in Fig.5.16(d) there is an example of occlusion. In this case, it is visible that the only variable that makes $HR(P1)_t$ to increase a little is AI. As soon as $P1$ becomes not occluded (Fig.5.16(e)) $HR(P1)_t$ increases (thanks to both the effect of not being occluded, having a visible frontal face and moving the arms). In Fig.5.16(f) $HR(P1)_t$ drops a little bit again as $P1$ stops moving the arms and is partially occluded. By looking at $P2$, in frames from Fig.5.16(h) to Fig.5.16(k), there are other examples of occlusion where $HR(P2)_t$ is 0. In the other examples, it is possible to see that the system correctly assigns higher values of human response to those situations where the two participants provided more feedback to the camera ($P2$ in frame Fig.5.16(f) and $P1$ in frames from Fig.5.16(h) to (k)). In frame Fig.5.16(e) it is also possible to see that $HR(P1)_t$ is higher than $HR(P2)_t$ as $P1$ is shaking his arms and looking towards the robot while $P2$ is closer to the camera but neither smiling nor shaking her arms. In this frame, $P1$ is located at 4.6 meters which is almost the d_{max} threshold defined for the configuration of the used system. This threshold is related to the lost of precision of the image resolution and stereo accuracy.

Depending on the application where the system is used, different measures based on HR can be computed, such as the global HR (represented

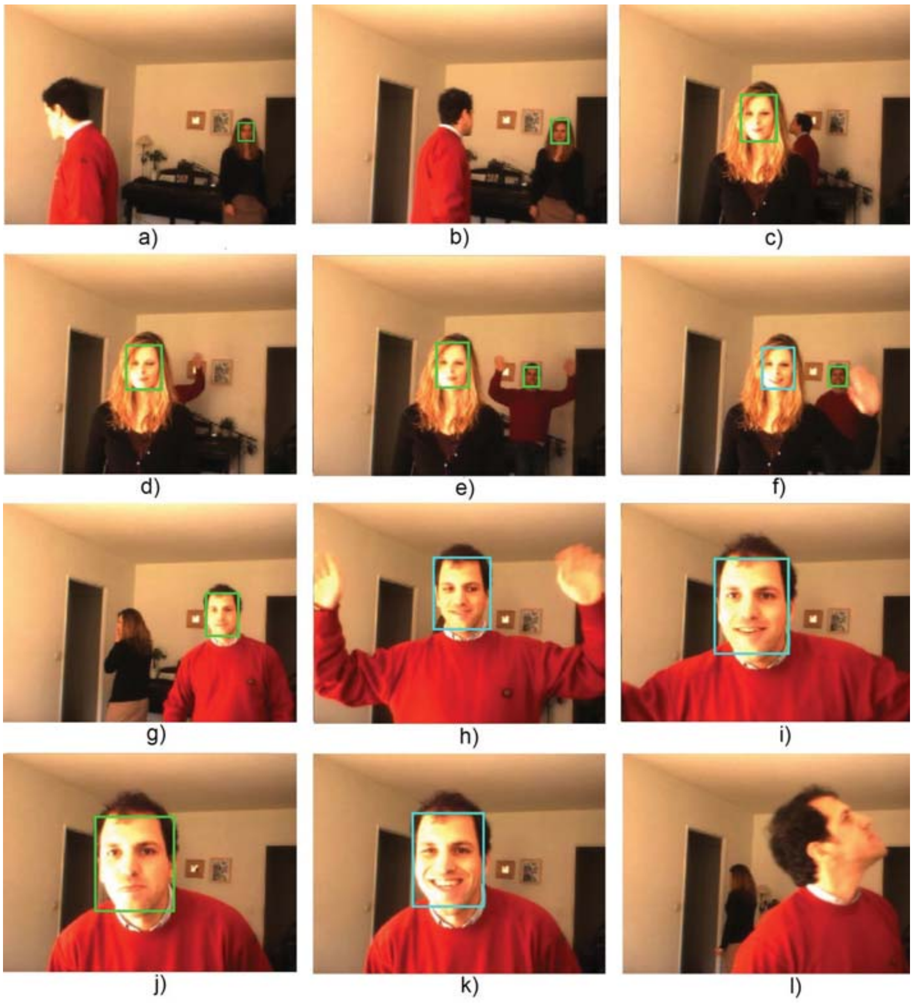


Figure 5.16: Different situations during an interaction between the robot and two people on its surroundings.

as $GHR(P)$), the average HR (represented as $\overline{HR(P)}$), the standard deviation of HR (represented as $\sigma(HR(P))$), the maximum HR (represented as $\max(HR(P))$) and the minimum HR (represented as $\min(HR(P))$). These values supply important information about the whole or a part of the interaction. The values for this experiment are presented on Table 5.3. This functionality could be applied to improve activities, and evaluate applications and people in a global way rather than only based on an instantaneous human response value.

Table 5.3: Statistical Information for HR(P1) and HR(P2)

Measure	P1	P2
$GHR(P)$	431.10	323.94
$\overline{HR(P)}$	0.56	0.36
$\sigma(HR(P))$	0.35	0.32
$\max(HR(P))$	1	1
$\min(HR(P))$	0	0

5.3 Summary and Final Remarks

In this Chapter, a fuzzy stereo-vision system able to detect and analyse the human response of users located in the surroundings of a social robot is presented. One of the system goals is to be as “natural” as possible and to avoid being user intrusive. Following this logic, a single stereo camera that has several similarities regarding the human vision system, is the only sensor used to capture information.

We observed that the system is able to detect different visual cues which were previously defined. As there are visual cues which are not easily detectable (mainly due to hardware constraints) at all distances, the system correctly selects those visual cues which can provide valuable information to the human response value. In addition, people can move and interact freely, and the system was able to smoothly select among different sources of information according to the distance of the person to the camera. The system was tested in simulated real life situations, where people participating in the experiments were asked to move and to act naturally. In addition, the method is fast enough to be used in real time applications.

The approach is based on FL. Thus, human response is computed by means of a Hierarchical Fuzzy System(s) (HFS) that is able to deal with the uncertainty and vagueness of the inputs, depending on the distance of the person. FL is a very helpful tool that has a well known efficacy treating uncertain and vague information. It also aids to deal with noisy data. In this kind of sensors, information supplied is commonly affected by errors, and therefore the use of FS is helpful when dealing with this problem. By setting up linguistic variables and “natural” alike rules, the problem is managed in an efficient way. Also, by using FS to represent knowledge, the understanding of the system is facilitated, as this kind of knowledge representation is similar to the way the human beings represent the knowledge. Moreover, it is not complex to change or add other variables, so making easy to improve or customise the system.

Chapter 6

Conclusions and Future Work

This thesis has presented contributions in different areas of the Soft Computing (SC), Computer Vision and Human-Robot Interaction (HRI) fields. Efforts have been focused in the problem of people detection and tracking which could be considered a first step before developing any other HRI techniques. Additionally, we have proposed a novel approach to detect different kinds of human responses interacting with a robot. This chapter aims at highlighting the main contributions and providing a summarising view of the work developed. Two main problems have been addressed in this Thesis: the problem of people detection and tracking (Chapter 3) and the recognition of people interest in interacting with a robotic agent (Chapters 4 and 5).

The main contributions of this PhD Thesis are:

- The development of a fast stereo tracking algorithm using a confidence measure. The confidence measure is employed to modify the probability distribution function employed for weighting the particles in the particle filtering algorithm. This proposal is robust and allows to manage the uncertainty associated to the disparity information.

- The development of a fuzzy stereo tracking algorithm. In this proposal not only the uncertainty associated to disparity information is managed. The managing of the vagueness associated to the rest of sources of information is considered too.

- A new fuzzy system that allows the visual detection of interaction demands. A level of interest is computed in realtime by a view based approach using Support Vector Machines.

- The proposal of a hierarchical fuzzy system to measure human response using stereo vision. The hierarchical fuzzy system is able to deal with the uncertainty and vagueness of the measures depending on the distance of the tracked person.

In Chapter 3, two people detection and tracking methods are presented, one which is based on a probabilistic approach (described in Section 3.1) and another based on a "possibilistic" approach (Section 3.2). Both of them combine multiple visual cues using a Particle Filter (PF). Both methods employ the concept of projection of a person which is comprised of two ellipses: one for the person head and another one for his/her the torso.

On the probabilistic approach, particles represent possible 3D positions for the model that are evaluated by examining their projection in the camera image. This method integrates depth, color and gradient information to perform a robust tracking. As depth information cannot be always extracted because of occlusions or absence of texture, the probabilistic method is able to deal with this problem by defining a certainty measure that indicates the degree of confidence in depth information. The confidence measure is employed to modify the probability distribution function employed for weighting the particles. The greater is the amount of disparity found, the greater is its contribution to the weight of the final particle and vice versa. In the extreme case of complete unavailability of disparity, the tracking is done using only colour and gradient information. The proposed algorithm does not only determine the 3D person position but also his/her head position in the camera image.

Several color-with-depth sequences have been employed in order to test the validity of our proposal. The sequences recorded show a varying number of people (from one up to four) interacting in a room. In the sequences, people perform different types of interactions: walk at different distances, shake hands, cross their paths, jump, run, embrace each other and even quickly swap their positions trying to confuse the system. The tracking errors have been calculated for different number of particles in order to determine the number of them that allows an appropriate trade-off between tracking error and computing time. The experimental results show that the proposed method is able to determine, in real-time, both the 3D position and the 2D head position in the camera image of a moving person despite of the presence of other people. Besides, the proposed method is able to deal with both partial and short-term total occlusion.

The “possibilistic” approach tries to overcome some situations which were not taken into account on the probabilistic approach. Firstly, using the previous approach, there are some cases in which the background color can be confused with the color model of the tracked objects. In this second approach, foreground and background information is taken into consideration which avoid certain confusing situations. Secondly, the probabilistic approach does not handle certain occlusion situations which are now taken into consideration. Thirdly, in the probabilistic approach, there is only one confidence measure based on the disparity information. In the second approach, not only disparity information is taken into consideration for computing confidence levels, but also the distance at which the person is located and the possibility of being occluded. We consider that if a person is partially or completed occluded, the confidence on disparity or on color information will change.

In addition to these advantages, a more elaborated people detection method is employed in order to reduce the amount of false positives. Furthermore, in the second approach the 2D tracking error is lower than on the first method.

The previously mentioned advantages are mainly due to the use of an increasing number of information cues in the second method. This is achieved with the help of Fuzzy Logic (FL) which has the ability of managing an increasing amount of information. In addition, the use of FL to compute the final weight of each particle brings different benefits compared to the probabilistic approach. First, by using probability models to evaluate particles, it is assumed that variables follow a probability distribution. That is, uncertainty could be modeled in a probabilistic approach by modifying the probabilistic distribution function using some parameter. Those assumptions sometimes are not exactly true or are hard to be modeled. Nevertheless, with FL it is possible to achieve the same goal in a more flexible way, without being restricted to particular aspects of the probability distributions. Secondly, FL easily allows to incrementally add other sources of information. By using linguistic variables and rules to express relationships the system becomes more understandable and similar to the way humans represent and deal with the knowledge.

On the other hand, there is the drawback of using more information cues which is directly related with the computing time. The computing time is higher than in the first method but the second method is able to deal with more complicated situations than the first one.

The second method has been experimentally compared to other well-known tracking methods. The results showed that our system managed to keep track of people, in the reference image, in most of the situations where other trackers fail. It was tested in simulated complex real life situations, where people were interacting freely and sometimes occluding each other. The second method proved to be also fast enough for detecting and tracking people simultaneously and therefore adequate to be used in real time applications.

In Chapter 4, a system for estimating the interest of the people in the surroundings of the robot and detecting motions related to attention demand and head nodding and shaking, is described. It uses Stereo Vision (SV), head pose estimation by Support Vector Machines (SVM) and FL. While a person is being tracked, the Fuzzy System(s) (FS) computes a level of possibility about the interest that this person has in interacting with the robot. This possibility value is based on the position of the person with respect to the robot, as well as on an estimation of the attention that the person pays to the robot. To examine the attention that a person pays to the robot the head pose of the person is analyzed in real time. This analysis is performed by a view based approach using SVM. Thanks to SVM, the head pose can be detected achieving a great percentage of success that does not depend on the morphological features of the heads. The employed FS was also able to detect accurately whenever the person was demanding attention by analyzing the movement of the arms. The system also achieved a good result concerning the detection of head movements such as head nodding and shaking.

In Chapter 5, the human response is computed without prioritizing people that are closer, more or less centered and/or facing the robot, regarding other people that are not fulfilling all and every condition. Now, we try to improve the human behavior analysis by defining a human response measure based on visual cues which have different importance levels depending on the distance of the person towards the robot. These visual cues are the attention that each person is paying to him (face pose regarding the robot), occlusion (considering that people will try to avoid obstacles when interacting with it), certain arms movements (the robot might ask people to interact by shaking their arms) and smile detection (it should provide a good feedback about each person satisfaction level according to those actions and messages transmitted by the robot). FL will be the tool used to fuse all these information cues as it allows the handling of the uncertainty and vagueness that come along with the information provided by the sensors. A Hierarchical Fuzzy System(s) (HFS) is proposed so that a high level FS handles the outputs of two low level FS depending on the distance information of the tracked person. Each low level FS is specialized in one specific distance situation of the person regarding the robot: close or far. The high level FS fuses the information supplied by the low level FS according to the distance value. This is done so that visual information is managed in a correct and useful way in order to measure the human response. Several final measures are defined based on the output of the proposed HFS such as the instantaneous human response $HR(P)$, the global human response $GHR(P)$ and the average value of $HR(P)$. By taking into consideration instantaneous, global and average values, one might analyze which aspects of the robot proposed activity might be improved or adapted. For instance, the robot may adapt its behavior whenever the feedback given by the users of the system is not satisfactory.

The system was tested in simulated real life situations, where people participating in the experiments were asked to move and to act naturally. Despite the high amount of data involved, we observed that the proposed algorithm is prepared to be used in real time environments and so allowing a natural interaction among all the people that participated in the experiments. In addition, the HFS correctly outputs the expected human response values on most of the frames.

In the different FS shown in this work, rules and linguistic variables have been defined experimentally. As a future work, the possibility of building a system capable of learning and adjusting these parameters automatically is being studied. Another possible improvement for future works is to use the feedback from users regarding the activities proposed by the social robot in order to improve those same activities and the interaction between the robot and users. Finally, the modularity of the system allows the incorporation of other information sources. Therefore, sound sensors and speech recognition techniques are being studied for incorporation in future works.

Bibliography

[AAOM12] A. Albiol, A. Albiol, J. Oliver, and J.M. Mossi. Who is who at different cameras: people re-identification using depth cameras. *Computer Vision, IET*, 6(5):378–387, 2012. [cited at p. 27]

[AC13] N. N. Singh A. Chatterjee, A. Rakshit. *Vision Based Autonomous Robot Navigation.* Springer Berlin Heidelberg, 2013. [cited at p. 6]

[AD12] I. Ali and M. N. Dailey. Multiple human tracking in high-density crowds. *Image and Vision Computing*, 30(12):966 – 977, 2012. [cited at p. 24]

[AG00a] E. Aguirre and A. González. Fuzzy behaviors for mobile robot navigation: Design, coordination and fusion. *International Journal of Approximate Reasoning*, 25:255–289, 2000. [cited at p. 11]

[AG00b] E. Aguirre and A. González. Integrating topological and geometrical world modeling for mobile robot navigation. In *I Workshop Hispano-Luso de Agentes Físicos*, pages 167–181, Tarragona, 2000. (In spanish). [cited at p. 11]

[AG02] E. Aguirre and A. González. Integrating fuzzy topological maps and fuzzy geometric maps for behavior-based robots. *International Journal of Intelligent Systems*, 17(3):333–368, 2002. [cited at p. 7]

[AGSG$^+$07] E. Aguirre, M. Garcia-Sílvente, A. González, R. Paúl, and R. Muñoz-Salinas. A fuzzy system for detection of interaction demanding and nodding assent based on stereo vision. *Journal of Physical Agents*, 1:15–25, 2007. [cited at p. xxiv, 28, 171]

[AK10] K. Ambrosch and W. Kubinger. Accurate hardware-based stereo vision. *Computer Vision and Image Understanding*, 114(11):1303 – 1316, 2010. [cited at p. 22]

[AKK08] S. Asteriadis, K. Karpouzis, and S. Kollias. A neuro-fuzzy approach to user attention recognition. In *Proceedings of the 18th international conference on Artificial Neural Networks, Part I*, ICANN '08, pages 927–936. Springer-Verlag, 2008. [cited at p. 27]

[Alb92] J. S. Albus. RCS: A reference model architecture for intelligent control. *j-COMPUTER*, 25(5):56–59, 1992. [cited at p. 9]

[Alb99] J. S. Albus. 4-d/rcs reference model architecture for unmanned ground vehicles. In *Proceedings of the SPIE*, volume 3693, pages 11–20, 1999. [cited at p. 9]

[Ark86] R. C. Arkin. Path planning for a vision-based autonomous robot. In *Proc. of the SPIE Conference on Mobile Robots*, pages 240–249, 1986. [cited at p. 10]

[Ark98] R. C. Arkin. *Behavior-Based Robotics.* The MIT Press, 1998. [cited at p. 2, 9]

[AS11] H. C. Akakin and B. Sankur. Robust classification of face and head gestures in video. *Image and Vision Computing*, 29(7):470–483, 2011. [cited at p. 28]

[AST11] A. Andreas, E.F. Soonggalon, and K. Tejawibawa. Developing a quadrupedal robot with speech recognition movement control. In *Instrumentation Control and Automation (ICA), 2011 2nd International Conference on*, pages 310–315, 2011. [cited at p. 17]

[ATKTJ00] M.-R. Akbarzadeh-T, K. Kumbla, E. Tunstel, and M. Jamshidi. *Soft computing for autonomous robotic systems.* Elsevier, 2000. [cited at p. 11]

[ATR97] F. Aherne, N. Thacker, and P. Rockett. The bhattacharyya metric as an absolute similarity measure for frequency coded data. *Kybernetica*, 32:1–7, 1997. [cited at p. 24, 35, 176]

[BBB$^+$98] H.-J. Boehme, A. Brakensiek, U.-D. Braumann, M. K.s, and H.-M. Gross. Neural architecture for gesture-based human-machine-interaction. *Lecture Notes in Computer Science*, 1371:219–232, 1998. [cited at p. 21]

[BBH03] M. Z. Brown, D. Burschka, and G. D. Hager. Advances in computational stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:993–1008, 2003. [cited at p. xxii, 6, 33, 169, 174]

[BCZ93] A. Blake, R. Curwen, and A. Zisserman. A framework for spatiotemporal control in the tracking of visual contours. *International Journal of Computer Vision*, 11:127–145, 1993. [cited at p. 67]

[BFJ$^+$05a] M. Bennewitz, F. Faber, D. Joho, M. Schreiber, and S. Behnke. Integrating vision and speech for conversations with multiple persons. In *IROS'05: Proceedings of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 2523–2528, 2005. [cited at p. xxiii, 16, 28, 169]

[BFJ$^+$05b] M. Bennewitz, F. Faber, D. Joho, M. Schreiber, and S. Behnke. Multimodal conversation between a humanoid robot and multiple persons. In *AAAI'05: Proceedings of the Workshop On Modular Construction of Human-Like Intelligence*, pages 40–47, 2005. [cited at p. 19]

[BGV92] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152. ACM Press, 1992. [cited at p. 39, 179]

[BH94] A. Baumberg and D. Hogg. An efficient method for contour tracking using active shape models. In *IEEE Workshop on Motion of Non-Rigid and Articulated Objects.*, pages 194–199, 1994. [cited at p. 67]

[BHD94] R. Beckers, O. E. Holland, and J. L. Deneubourg. From local actions to global tasks: Stigmergy and collective robotics. In *Artificial Life IV*, pages 181–189. MIT Press, 1994. [cited at p. 13]

[Bir98] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *IEEE Conference on Computer Vision and Pattern Recognition (1998)*, pages 232–237, 1998. [cited at p. 35, 67, 84, 175, 176]

[BJ11] P. Benavidez and M. Jamshidi. Mobile robot navigation and target tracking system. In *System of Systems Engineering (SoSE), 2011 6th International Conference on*, pages 299–304, 2011. [cited at p. 18]

[BK08] G. Bradski and A. Kaehler. *Learning OpenCV. Computer Vision with the OpenCV.* O'Reilly, 2008. [cited at p. 72]

[Blo08] I. Bloch. Defining belief functions using mathematical morphology - application to image fusion under imprecision. *International Journal of Approximate Reasoning*, 48:437–465, 2008. [cited at p. 25]

[BPM$^+$08] J.N. Bailenson, E.D. Pontikakis, I.B. Mauss, J.J. Gross, M.E. Jabon, C. Hutcherson, C. Nass, and O. John. Real-time classification of evoked emotions using facial feature tracking and physiological responses. *International Journal Human-Computer Studies*, 66(5):303–317, 2008. [cited at p. 28]

[Bre02] C. Breazeal. *Designing Sociable Robots.* MIT Press, 2002. [cited at p. 15, 17, 19]

[Bre03] C. Breazeal. Towards sociable robots. *Robotics and Autonomous Systems*, 42(3):167–175, March 2003. [cited at p. 13]

[Bro85] R.A. Brooks. A layered intelligent control system for a mobile robot. In *Third International Symposium of Robotics Research*, pages 1–8, Gouvieux, France, 1985. [cited at p. 9]

[Bro91] R. A. Brooks. Intelligence without reason. In John Myopoulos and Ray Reiter, editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 569–595, Sydney, Australia, 1991. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA. [cited at p. 1]

[BS99] C. Breazeal and B. Scassellati. How to build robots that make friends and influence people. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 858–863, 1999. [cited at p. 20]

[BS03] Z. Bien and W. Song. Blend of soft computing techniques for effective human-machine interaction in service robotic systems. *Fuzzy Sets and Systems*, 134(1):5–25, 2003. [cited at p. 21]

[BV10] J. Biswas and M. Veloso. Wifi localization and navigation for autonomous indoor mobile robots. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 4379–4384, 2010. [cited at p. 9]

[CDL10] G. Cielniak, T. Duckett, and A. J. Lilienthal. Data association and occlusion handling for vision-based people tracking by mobile robots. *Robotics and Autonomous Systems*, 58(5):435 – 443, 2010. [cited at p. 24]

[CHF08] H.Y. Chen, C.L. Huang, and C.M. Fu. Hybrid-boost learning for multi-pose face detection and facial expression recognition. *Pattern Recognition*, 41(3):1173–1185, 2008. [cited at p. 16, 28]

[CHX$^+$10] Ling Cai, Lei He, Yiren Xu, Yuming Zhao, and Xin Yang. Multi-object detection and tracking by stereo vision. *Pattern Recognition*, 43(12):4028 – 4041, 2010. [cited at p. 24]

[CL11] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. [cited at p. 39, 115, 179]

[CR00] D. Comaniciu and V. Ramesh. Mean shift and optimal prediction for efficient object tracking. In *IEEE International Conference on Image Processing (2000)*, volume 3, pages 70–73, 2000. [cited at p. 24, 35, 87, 175, 176]

[CV95] C. Cortes and V. Vapnik. Support-vector networks. In *Machine Learning*, pages 273–297, 1995. [cited at p. 39, 179]

[CV12] G. Csaba and Z. Vamossy. Fuzzy based obstacle avoidance for mobil robots with kinect sensor. In *Logistics and Industrial Informatics (LINDI), 2012 4th IEEE International Symposium on*, pages 135–144, 2012. [cited at p. 18]

[Dau95] K. Dautenhahn. Getting to know each other-artificial social intelligence for autonomous robots. *Robotics and Autonomous Systems*, (16):333–356, 1995. [cited at p. 14]

[Dau97] K. Dautenhahn. I could be you — the phenomenological dimension of social understanding. *Cybernetics and Systems*, 25(8):417–453, 1997. [cited at p. 15]

[Dau98] K. Dautenhahn. The art of designing socially intelligent agents - science, fiction and the human in the loop. *Applied Artificial Intelligence Journal, Special Issue on Socially Intelligent Agents*, 12:573–617, 1998. [cited at p. 14, 16]

[Dau02] K. Dautenhahn. Design spaces and niche spaces of believable social robots. In *Proceedings. 11th IEEE International Workshop on Robot and Human Interactive Communication*, pages 192–197, 2002. [cited at p. 15]

[Dau03] K. Dautenhahn. Roles and functions of robots in human society: implications from research in autism therapy. *Robotica*, 21(4):443–452, 2003. [cited at p. 15]

[DB99] K. Dautenhahn and A. Billard. Bringing up robots or - the psychology of socially intelligent robots: from theory to implementation. In *Proceedings of the Third International Conference on Autonomous Agents (Agents'99)*, pages 366–367. ACM Press, 1999. [cited at p. 15]

[DD96] M. Reinfrank D. Driankov. *An Introduction to Fuzzy Control*. Springer, 1996. [cited at p. 39, 180]

[DDCF01] T. Darrell, D. Demirdjian, N. Checka, and P. Felzenszwalb. Plan-view trajectory estimation with dense stereo background models. In *Eighth IEEE International Conference on Computer Vision (ICCV 2001)*, volume 2, pages 628 – 635, 2001. [cited at p. 23]

[DGF$^+$90] J. L. Deneubourg, S. Goss, N. Franks, A. Sendova-Franks, C. Detrain, and L. Chrétien. The dynamics of collective sorting robot-like ants and ant-like robots. In *Proceedings of the first international conference on simulation of adaptive behavior on From animals to animats*, pages 356–363. MIT Press, 1990. [cited at p. 13]

[DGHW00] T. Darrell, G. Gordon, M. Harville, and J. Woodfill. Integrated Person Tracking Using Stereo, Color, and Pattern Detection. *Int. Journ. Computer Vision*, 37:175–185, 2000. [cited at p. 24]

[DJ96] H. S. Dulimarta and A. K. Jain. A client/server control architecture for robot navigation. *Pattern Recognition*, 29(8):1259–1284, 1996. [cited at p. 9]

[DK02] G. N. DeSouza and A. C. Kak. Vision for Mobile Robot Navigation: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:237–267, 2002. [cited at p. 6]

[DO02] K. Dautenhahn and B. Ogden. From embodied to socially embedded agents implications for interaction-aware robots. *Cognitive Systems Research*, 3:397–428, 2002. [cited at p. 14]

[Duf03] B. Duffy. Anthropomorphism and the social robot. *Special Issue on Socially Interactive Robots, Robotics and Autonomous Systems*, 42(3):177–190, 2003. [cited at p. 15]

[EKB98] C. Eveland, K. Konolige, and R.C. Bolles. Background Modelling for Segmentatation of Vide-Rate Stereo Sequences. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 266–271, 1998. [cited at p. 23]

[Elf89] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *IEEE Computer Magazine, Special Issue on Autonomous Intelligent Machines*, 22(6):46–57, 1989. [cited at p. 6]

[ETNMT11] S.A.-L. El-Teleity, Z.B. Nossair, H.M.A.-K. Mansour, and A. TagElDein. Fuzzy logic control of an autonomous mobile robot. In *Methods and Models in Automation and Robotics (MMAR), 2011 16th International Conference on*, pages 188–193, 2011. [cited at p. 22]

[FLW$^+$12] Nai-Hong Fang, I-Hsum Li, Wei-Yen Wang, Lian-Wang Lee, and Yi-Hsing Chien. Research and design of control system for a tracked robot with a kinect sensor. In *System Science and Engineering (ICSSE), 2012 International Conference on*, pages 217–222, 2012. [cited at p. 18]

[FND03] T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42:143–166, 2003. [cited at p. xxii, 15, 24, 169]

[Fog99] B. J. Fogg. Persuasive technologie. *Commun. ACM*, 42(5):26–29, 1999. [cited at p. 15]

[FSA11] J. Foytik, P. Sankaran, and V. Asari. Tracking and recognizing multiple faces using kalman filter and modularpca. *Procedia Computer Science*, 6:256 – 261, 2011. [cited at p. 25]

[FTB01] T. W. Fong, C. Thorpe, and C. Baur. Collaboration, dialogue, and human-robot interaction. In *Proceedings of the 10th International Symposium of Robotics Research, Lorne, Victoria, Australia*. Springer-Verlag, November 2001. [cited at p. 20]

[Ful10] R. Fuller. A short survey of fuzzy reasoning methods - tutorial, http://uni-obuda.hu/users/fuller.robert/fuzzy-reasoning.pdf, 2010. [cited at p. 39, 180]

[FvD82] J.D. Foley and A. van Dam. *Fundamentals of Interactive Computer Graphics*. Addison Wesley, 1982. [cited at p. 35, 176]

[Gat91] E. Gat. *Reliable Goal-Directed Reactive Control of Autonomous Mobile Robots*. PhD thesis, Virginia Polytechnic Institute, 1991. [cited at p. 10]

[Gav99] D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding: CVIU*, 73(1):82–98, 1999. [cited at p. 17]

[GBL^{+}11] N. Greggio, A. Bernardino, C. Laschi, J. Santos-Victor, and P. Dario. Real-time 3d stereo tracking and localizing of spherical objects with the icub robotic platform. *J. Intell. Robotics Syst.*, 63(3-4):417–446, September 2011. [cited at p. 22]

[GK04] D. Grest and R. Koch. Realtime multi-camera person tracking for immersive environments. In *IEEE Sixth Workshop on Multimedia Signal Processing (2004)*, pages 387–390, 2004. [cited at p. 23, 35, 175]

[GNS^{+}02] S. Ghidary, Y. Nakata, H. Saito, M. Hattori, and T. Takamori. Multimodal interaction of human and home robot in the context of room map generation. *Autonomous Robots*, 13:169–184, 2002. [cited at p. xxiii, 16, 169]

[GS95] N. Gordon and D. Salmand. Bayesian state estimation for tracking and guidance using the bottstrap filter. *Journal of Guidance, Control and Dynamics*, 18:1434–1443, 1995. [cited at p. 25]

[GS99] J. Gasós and A. Saffiotti. Using fuzzy sets to represent uncertain spatial knowledge in autonomous robots. *Spatial Cognition and Computation*, 1(3):205–226, 1999. [cited at p. 11]

[GSFVGG97] M. Garcia-Silvente, J. Fdez-Valdivia, J.A. Garcia, and A. Garrido. A new edge detector integrating scale-spectrum information. *Image and Vision Computing*, 15(12):913–923, 1997. [cited at p. 11]

[Har04] M. Harville. Stereo person tracking with adaptive plan-view templates of height and occupancy statistics. *Image and Vision Computing*, 2:127–142, 2004. [cited at p. 23, 25]

[HCPW03] Chia-Chiang Ho, Wen-Huang Cheng, Ting-Jian Pan, and Ja-Ling Wu. A user-attention based focus detection framework and its applications. In *International Conference on Information, Communications Signal Processing Fourth IEEE Pacific-Rim Conference On Multimedia*, pages 1315–1319, 2003. [cited at p. 27]

[HD89] G. Henry and Dunteman. *Principal Components Analysis.* SAGE Publications, 1989. [cited at p. 36, 177]

[HGW01] M. Harville, G. Gordon, and J. Woodfill. Foreground segmentation using adaptive mixture models in color and depth. In *IEEE Workshop on Detection and Recognition of Events in Video (2001)*, pages 3–11, 2001. [cited at p. 23, 73]

[HH12] F. Huo and E. A. Hendriks. Multiple people tracking and pose estimation with occlusion estimation. *Computer Vision and Image Understanding*, 116(5):634 – 647, 2012. [cited at p. 24]

[HLK09] S. Hong, H. Lee, and E. Kim. A new probabilistic fuzzy model: Fuzzification-maximization (fm) approach. *International Journal of Approximate Reasoning*, 50:1129–1147, 2009. [cited at p. 25]

[HM03] N. Hirai and H. Mizoguchi. Visual tracking of human back and shoulder for person following robot. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (2003)*, volume 1, pages 527–532, 2003. [cited at p. xxi, 168]

[HMW$^+$09] F. Hegel, C. Muhl, B. Wrede, M. Hielscher-Fastabend, and G. Sagerer. Understanding social robots. In *Advances in Computer-Human Interactions, 2009. ACHI '09. Second International Conferences on*, pages 169–174, 2009. [cited at p. 12]

[HOP11] D.H. Heo, A. Oh, and T.H. Park. A localization system of mobile robots using artificial landmarks. In *Automation Science and Engineering (CASE), 2011 IEEE Conference on*, pages 139–144, 2011. [cited at p. 9]

[HST01] A. Howard, H. Seraji, and E. Tunstel. A Rule-Based Fuzzy Traversability Index for Mobile Robot Navigation. In *IEEE International Conference on Robotics and Automation*, pages 3067–3071, 2001. [cited at p. 11]

[HW] O. Holland and G. Walter. The pioneer of real artificial life. In *Proceedings of the International Workshop on Artificial Life, MIT Press*, pages 34–44, Cambridge, MA. [cited at p. 13]

[IB98] M. Isard and A. Blake. Condensation-conditional density propagation for visual trackings. *International Journal of Computer Vision*, 29:5–28, 1998. [cited at p. 23, 25, 80]

[IBP06] R.T. Iqbal, C. Barbu, and F. Petry. Fuzzy component based object detection. *International Journal of Approximate Reasoning*, 45:546–563, 2006. [cited at p. 25]

[ISH12] M. Ilbeygi and H. Shah-Hosseini. A novel fuzzy facial expression recognition system based on facial feature extraction from color face images. *Engineering Applications of Artificial Intelligence*, 25(1):130–146, 2012. [cited at p. 29]

[JF93] J.L. Jones and A. M. Flynn. *Mobile Robots. Inspiration to Implementation.* A K Peters, 1993. [cited at p. 3, 4, 7]

[JF98] L. C. Jain and T. Fukuda. *Soft Computing for Intelligent Robotic Systems.* Physica-Verlag, 1998. [cited at p. 11]

[Kae] Steven D. Kaehler. Fuzzy logic tutorial, http://www.seattlerobotics.org/encoder/mar98/fuz/flindex.html. [cited at p. 39, 180]

[Kai67] T. Kailath. The Divergence and Bhattacharyya Distance Measures in Signal Selection. *IEEE Transactions on Communication Technology*, 15:52 – 60, 1967. [cited at p. 35, 176]

[KB05] H. Kamel and W. Badawy. Fuzzy-logic-based particle filter for tracking a maneuverable target. In *48th Midwest Symposium on Circuits and Systems (2005)*, volume 2, pages 1537–1540, 2005. [cited at p. 26]

[KBD05] Z. Khan, T. Balch, and F. Dellaert. Mcmc-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:1805–1819, 2005. [cited at p. 26]

[KC03] D. Kulic and E.A. Croft. Estimating intent for human robot interaction. In *International Conference on Advanced Robotics*, pages 810–815, 2003. [cited at p. 16, 21]

[KG01] L. Kopp and P. Gardenfors. Attention as a minimal criterion of intentionality in robotics. In *Lund University of Cognitive Studies*, volume 89, 2001. [cited at p. 20]

[Kit96] G. Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5:1–25, 1996. [cited at p. 25]

[KjB97] L. Kil-jae and Z. Bien. A model-based machine vision system using fuzzy logic. *International Journal of Approximate Reasoning*, 16:119–135, 1997. [cited at p. 25]

[KLM10] J. Klein, C. Lecomte, and P. Miché. Hierarchical and conditional combination of belief functions induced by visual tracking. *International Journal of Approximate Reasoning*, 51:410–428, 2010. [cited at p. 25]

[KM97] K. Konolige and K. Myers. *In Artificial Intelligence and Mobile Robots:Case Studies of Successful Robot Systems.* The MIT Press, 1997. [cited at p. 11]

[KPKK11] Seong-Yong Koo, Kiru Park, Hyun Kim, and Dong-Soo Kwon. A dual-layer user model based cognitive system for user-adaptive service robots. In *RO-MAN, 2011 IEEE*, pages 59–64, 2011. [cited at p. 20]

[KV10] J. Krejsa and S. Vechet. Odometry-free mobile robot localization using bearing only beacons. In *Power Electronics and Motion Control Conference (EPE/PEMC), 2010 14th International*, pages T5–40–T5–45, 2010. [cited at p. 8]

[KW94] D. Kortenkamp and T. Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proc. of the Twelfth National Conf. on AI (AAAI-94)*, pages 979–984, Menlo Park, Calif., 1994. [cited at p. 7]

[KY01a] H. Kozima and H. Yano. In search of ontogenetic prerequisites for embodied social intelligence. In *Proceedings of the Workshop on Emergence and Development on Embodied Cognition*, pages 30–34, International Conference on Cognitive Science, 2001. [cited at p. 20]

[KY01b] H. Kozima and H. Yano. A robot that learns to communicate with human caregivers. In *Proceedings of International Workshop on Epigenetic Robotics*, pages 47–52, 2001. [cited at p. 20]

[LDX11] Shuang Liu, Jie Dong, and Xin Xing. Shape of object recognition based on rs-ann for mobile robot. In *Mechatronic Science, Electric Engineering and Computer (MEC), 2011 International Conference on*, pages 193–196, 2011. [cited at p. 21]

[LHH12] Taegyu Lim, Bohyung Han, and Joon H. Han. Modeling and segmentation of floating foreground and background in videos. *Pattern Recognition*, 45(4):1696 – 1706, 2012. [cited at p. 23]

[LJW98] W. Li, X. Jiang, and Y. Wang. Road recognition for vision navigation of an autonomous vehicle by fuzzy reasoning. *Fuzzy Sets and Systems*, 93:275–280, 1998. [cited at p. 12]

[LKH$^+$03] S. Lang, M. Kleinehagenbrock, S. Hohenner, J. Fritsch, G. A. Fink, and G. Sagerer. Providing the basis for human-robot-interaction: a multi-modal attention system for a mobile robot. In *ICMI '03: Proceedings of the 5th international conference on Multimodal interfaces*, pages 28–35. ACM Press, 2003. [cited at p. 19]

[LM02] R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *IEEE Conf. on Image Processing*, pages 900–903, 2002. [cited at p. 72]

[Lor95] T. Loren. Overview of human-computer collaboration. *Knowledge-Based Systems*, 8(23):67–81, 1995. [cited at p. 19]

[LS00] C. Lisetti and D. Schiano. Automatic facial expression interpretation: Where human-computer interaction, artificial intelligence and cognitive science intersect. *Pragmatics and Cognition, Special Issue on Facial Information Precessing and Multidisciplinary Perpective,*, 8(1):185–235, 2000. [cited at p. 19]

[LS09] T. Lukasiewicz and U. Straccia. Description logic programs under probabilistic uncertainty and fuzzy vagueness. *International Journal of Approximate Reasoning*, 50:837–853, 2009. [cited at p. 25]

[LSA10] M. Luber, L. Spinello, and K. Oliver Arras. People tracking in rgb-d data with on-line boosted target models. In *IROS 2011*, pages 3844–3849, 2010. [cited at p. 27]

[LWT03] W. Liang, H. Weiming, and L. Tieniu. Recent developments in human motion analysis. *Pattern Recognition*, 36:585–601, 2003. [cited at p. 17]

[LYTZ13] H. Liang, J. Yuan, D. Thalmann, and Z. Zhang. Model-based hand pose estimation via spatial-temporal hand parsing and 3d fingertip localization. *The Visual Computer*, 29(6-8):837–848, 2013. [cited at p. 29]

[LZW$^+$09] Y. Lin, W.J. Zhang, C. Wu, G. Yang, and J. Dy. A fuzzy logics clustering approach to computing human attention allocation using eyegaze movement cue. *International Journal Human-Computer Studies*, 67(5):455–463, 2009. [cited at p. 28]

[MA07] R. L. Mandryk and M. S. Atkins. A fuzzy physiological approach for continuously modeling emotion during interaction with play technologies. *International Journal Human-Computer Studies*, 65(4):329–347, 2007. [cited at p. 28]

[MAF$^+$99] T. Matsui, H. Asoh, J. Fry, Y. Motomura, F. Asano, T. Kurita, I. Hara, and N. Otsu. Integrated natural spoken dialogue system of jijo-2 mobile robot for office services. In *AAAI/IAAI*, pages 621–627, 1999. [cited at p. 17]

[Mat95] M. Mataric. Issues and approaches in the design of collective autonomous agents. *Robotics and Autonomous Systems*, 16:321–331, December 1995. [cited at p. 13]

[MB10] Manuel Mucientes and Alberto Bugarin. People detection through quantified fuzzy temporal rules. *Pattern Recognition*, 43(4):1441 – 1453, 2010. [cited at p. 25]

[MBMM13] Matteo Munaro, Gioia Ballin, Stefano Michieletto, and Emanuele Menegatti. 3d flow estimation for human action recognition from colored point clouds. *Biologically Inspired Cognitive Architectures - Availabe online*, (0):–, 2013. [cited at p. 29]

[MH99] Y. Marom and G. Hayes. Preliminary approaches to attention for social learning. In *Proceedings of ACAI'99 Workshop on Biologically Inspired Machine Learning*, pages 8–18, 1999. [cited at p. 20]

[MH01] Y. Marom and G. Hayes. Attention and social situatedness for skill acquisition. In *Proceedings of the First International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*, 2001. [cited at p. 20]

[MI] S. N. H. Mirza and E. Izquierdo. Examining visual attention: A method for revealing users' interest for images on screen. In *QoMEX*, pages 207–212. [cited at p. 27]

[Mic10] Microsoft. Kinect official webpage, http://www.xbox.com/en-gb/kinect, 2010. [cited at p. 18]

[MJK12] Y. Motai, S. Kumar Jha, and D. Kruse. Human tracking from a mobile agent: Optical flow and kalman filter arbitration. *Signal Processing: Image Communication*, 27(1):83 – 95, 2012. [cited at p. 25]

[MK10] D. Maier and A. Kleiner. Improved gps sensor model for mobile robots in urban terrain. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 4385–4390, 2010. [cited at p. 8]

[MM13] G. Mastorakis and D. Makris. Fall detection system using kinect's infrared sensor. *Journal of Real-Time Image Processing*, 2013. [cited at p. 29]

[MMN89] R. Mohan, G. Medioni, and R. Nevatia. Stereo error detection, correction, and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:113–120, 1989. [cited at p. 34, 175]

[MS12] V. Maroulas and P. Stinis. Improved particle filters for multi-target tracking. *Journal of Computational Physics*, 231(2):602 – 611, 2012. [cited at p. 26]

[MSAGS07] R. Muñoz-Salinas, E. Aguirre, and M. García-Silvente. People detection and tracking using stereo vision and color. *Image and Vision Computing*, 25:995–1007, 2007. [cited at p. 25]

[MSAGSP07] R. Muñoz-Salinas, E. Aguirre, M. García-Silvente, and R. Paúl. A new person tracking method for human-robot interaction intended for mobile devices. *MICAI 2007: Advances in Artificial Intelligence Lecture Notes in Computer Science*, 4827:747–757, 2007. [cited at p. xxiv, 171]

[MSMCMCCP09] R. Muñoz-Salinas, R. Medina-Carnicer, F.J. Madrid-Cuevas, and A. Carmona-Poyato. Multi-camera people tracking using evidential filters. *International Journal of Approximate Reasoning*, 50:732–749, 2009. [cited at p. 25]

[MTACS02] F. Moreno, A. Tarrida, J. Andrade-Cetto, and A. Sanfeliu. 3d real-time head tracking fusing color histograms and stereovision. In *International Conference on Pattern Recognition (2002)*, pages 368–371, 2002. [cited at p. 25]

[MWDM98] M. Mataric, M. Williamson, J. Demiris, and A. Mohan. Behavior-based primitives for articulated control. In *Proceedings, From Animals to Animats 5, Fifth International Conference on Simulation of Adaptive Behavior (SAB-98)*, pages 165–170, Zurich, Switzerland, august 1998. [cited at p. 9]

[NBG$^+$99] I. Nourbakhsh, J. Bobenage, S. Grange, R. Lutz, R. Meyer, and A. Soto. An affective mobile robot educator with a full-time job. *Artificial Intelligence*, 114(1-2):95–124, 1999. [cited at p. 15]

[NG10] Lazaros Nalpantidis and Antonios Gasteratos. Stereo vision for robotic applications in the presence of non-ideal lighting conditions. *Image and Vision Computing*, 28(6):940 – 951, 2010. [cited at p. 23]

[NKDdW09] M. Nachtegael, E. Kerre, S. Damas, and D. Van der Weken. Special issue on recent advances in soft computing in image processing. *International Journal of Approximate Reasoning*, 50:1–2, 2009. [cited at p. 25]

[NKMG03] K. Nummiaro, E. Koller-Meier, and L. Van Gool. An adaptive color-based particle filter. *Image and Vision Computing*, 21:99–110, 2003. [cited at p. 24, 35, 68, 87, 175, 176]

[NS76] A. Newell and H.A. Simon. Computer science as empirical enquiry. *Communications of the ACM*, 19:113–126, 1976. [cited at p. 9]

[OKO^{+}12] S. Obdrzalek, G. Kurillo, F. Ofli, R. Bajcsy, E. Seto, H. Jimison, and M. Pavel. Accuracy and robustness of kinect pose estimation in the context of coaching of elderly population. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 1188–1193, 2012. [cited at p. 29]

[Oku01] H. Okuno. Human-robot interaction through real-time auditory and visual multipletalker tracking. pages 1402–1409, 2001. [cited at p. 17]

[OnPS06] V. Matellán Olivera, J. M. Ca nas Plaza, and O. Serrano Serrano. Wifi localization methods for autonomous robots. *Robotica*, 24(4):455–461, July 2006. [cited at p. 9]

[ope10] openNI. openni - the standard for framework for 3d sensing, http://www.openni.org/, 2010. [cited at p. 18]

[OTdF^{+}04] K. Okuma, A. Taleghani, D. de Freitas, J.J. Little, and D.G. Lowe. A boosted particle filter: multi target detection and tracking. *Lectures Notes in Computer Science*, 3021:28–39, 2004. [cited at p. 26]

[OUV98] G. Oriolo, G. Ulivi, and M. Vendittelli. Real-time map building and navigation for autonomous robots in unknown environments. *IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics*, 28(3):316–333, 1998. [cited at p. 11]

[PAGSMS12] R. Paúl, E. Aguirre, M. García-Silvente, and R. Muñoz-Salinas. A new fuzzy based algorithm for solving stereo vagueness in detecting and tracking people. *International Journal of Approximate Reasoning*, 53:693–708, 2012. [cited at p. xxiv, 17, 171]

[PATF13] D. Perdomo, J. B. Alonso, C. M. Travieso, and M. A. Ferrer. Automatic scene calibration for detecting and tracking people using a single camera. *Engineering Applications of Artificial Intelligence*, 26(2):924–935, 2013. [cited at p. 23]

[PC98] E. Paulos and J. Canny. Designing personal tele-embodiment. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 4, pages 3173–3178, 1998. [cited at p. 15]

[Pea01] P. Persson and et al. Understanding socially intelligent agents - a multilayered phenomenon. *IEEE Transactions on SMC*, 31(5):349–360, 2001. [cited at p. 15]

[Pee03] Peter Peer. Cvl face database, 2003. [cited at p. 116]

[PL11] C.B. Park and S.W. Lee. Real-time 3d pointing gesture recognition for mobile robots with cascade hmm and particle filter. *Image Vision Computing*, 29(1):51–63, 2011. [cited at p. 16, 18, 20]

[PMC12] F. Poiesi, R. Mazzon, and A. Cavallaro. Multi-target tracking on confidence maps: an application to people tracking. *Computer Vision and Image Understanding*, (0):–, 2012. [cited at p. 26]

[PMP$^+$03] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, and S. Thrun. Towards robotic assistants in nursing homes: Challenges and results. *Special issue on Socially Interactive Robots, Robotics and Autonomous Systems*, 42(3 - 4):271 – 281, 2003. [cited at p. 15]

[PMRR00] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss. The feret evaluation methodology for face recognition algorithms. volume 22, pages 1090–1104, 2000. [cited at p. 116]

[Pri05] Primesense. Primesense official webpage, http://www.primesense.com/, 2005. [cited at p. 18]

[PSH97] V. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997. [cited at p. 18]

[PWHR98] P.J. Phillips, H. Wechsler, J. Huang, and P. Rauss. The feret database and evaluation procedure for face recognition algorithms. *Image and Vision Computing*, 16(5):295–306, 1998. [cited at p. 116]

[RA90] J.J. Rodriguez and J.K. Aggarwal. Stochastic analysis of stereo quantization error. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:467–470, 1990. [cited at p. 34, 175]

[RBMHMU12] J.A. Rivera-Bautista, A. Marin-Hernandez, and L.F. Marin-Urias. Using color histograms and range data to track trajectories of moving people from a mobile robot platform. In *Electrical Communications and Computers (CONIELECOMP), 2012 22nd International Conference on*, pages 288–293, 2012. [cited at p. 27]

[RC08] R. E. Woods R. C.Gonzalez. *Digital Image processing*. Prentice Hall, 2008. [cited at p. 5]

[Res01] S. Restivo. Bringing up and booting up: Social theory and the emergence of socially intelligent robot. In *IEEE International Conference on Systems, Man, and Cybernetics*, volume 4, pages 2110–2117, 2001. [cited at p. 16]

[Res05] Point Grey Research. Point grey research, bumblebee 1 stereo camera, http://www.ptgrey.com/products/bumblebee/bumblebee.pdf, 2005. [cited at p. 31, 77, 173]

[Res10] Point Grey Research. Triclops sdk, http://www.ptgrey.com/products/triclopssdk/index.asp, 2010. [cited at p. 22, 33, 174]

[Rob] ActivMedia Robotics. Performance peoplebot robot, http://www.mobilerobots.com/researchrobots/peoplebot.aspx. [cited at p. 31, 173]

[Saf97] A. Saffiotti. The uses of fuzzy logic in autonomous robot navigation. *Soft Computing*, 1:180–197, 1997. [cited at p. 2, 11]

[SAJ+13] L. Susperregi, A. Arruti, E. Jauregi, B. Sierra, J.M. Martínez-Otzeta, E. Lazkano, and A. Ansuategui. Fusing multiple image transformations and a thermal sensor with kinect to improve person detection ability. *Engineering Applications of Artificial Intelligence - Available Online*, (0):–, 2013. [cited at p. 27]

[SAS01] D. Spiliotopoulos, I. Androutsopoulos, and C. D. Spyropoulos. Human-robot interaction based on spoken natural language dialogue. In *in: Proceedings of the European Workshop on Service and Humanoid Robots*, pages 25–27, 2001. [cited at p. 17]

[SB10] Y. Sun and L. Bentabet. A particle filtering and dsmt based approach for conflict resolving in case of target tracking with multiple cues. *Journal of Mathematical Imaging and Vision*, 36:159–167, 2010. [cited at p. 24]

[SBOGP08] K. Smith, S. O. Ba, JM. Odobez, and D. Gatica-Perez. Tracking the visual focus of attention for a varying number of wandering people. *IEEE Transactions Pattern Analysis Machine Intelligence*, 30(7):1212–1229, 2008. [cited at p. 27]

[Sca01] B. Scassellati. *Foundations for a theory of mind for a humanoid robot*. PhD thesis, 2001. Supervisor-Rodney Brooks. [cited at p. 15]

[Sca03] B. Scassellati. Investigating models of social development using a humanoid robot. In *Proceedings of the International Joint Conference on Neural Networks*, volume 4, pages 2704–2709, 2003. [cited at p. 20]

[SCFERB+09] C. Solana-Cipres, G. Fernandez-Escribano, L. Rodriguez-Benitez, J. Moreno-Garcia, and L. Jimenez-Linares. Real-time moving object segmentation in h.264 compressed domain based on approximate reasoning. *International Journal of Approximate Reasoning*, 51:99–114, 2009. [cited at p. 25]

[Sch94] A. Schultz. Learning robot behaviors using genetic algorithms. In *Intelligent Automation and Soft Computing: Trends in Research, Development, and Applications*, pages 607–612, 1994. [cited at p. 21]

[SCSD09] R.E.O. Schultz, T.M. Centeno, G. Selleron, and M.R. Delgado. A soft computing-based approach to spatio-temporal prediction.

International Journal of Approximate Reasoning, 50:3–20, 2009. [cited at p. 25]

[SELG10] K. Schindler, A. Ess, B. Leibe, and L. Van Gool. Automatic detection and tracking of pedestrians from a moving stereo rig. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(6):523 – 537, 2010. [cited at p. 22]

[SF13] B. John Southwell and Gu Fang. Human object recognition using colour and depth information from an rgb-d kinect sensor. *International Journal of Advanced Robotic Systems*, 10:1–8, 2013. [cited at p. 27]

[SH94] F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. In *Proceedings of the Second IEEE Workshop on Applications of Computer Vision*, pages 138–142, 1994. [cited at p. 116]

[Sim13] C. Simpkins. Principle components analysis - a short primer by chris simpkins, http://www.cc.gatech.edu/ simpkins/courses/cs7641/pca-primer.txt, 2013. [cited at p. 36]

[SKKB01] W. Song, D. Kim, J. Kim, and Z. Bien. Visual servoing for a user's mouth with effective intention reading in a wheelchair-based robotic arm. In *ICRA*, pages 3662–3667, 2001. [cited at p. xxiii, 16, 28, 169]

[SM04] H. Suzuki and M. Minami. Real-time face detection using hybrid ga based on selective attention. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1329–1334, 2004. [cited at p. 27]

[SM11] T. T. Santos and C. H. Morimoto. Multiple camera people detection and tracking using support integration. *Pattern Recognition Letters*, 32(1):47 – 55, 2011. [cited at p. 23]

[SMC05] L. Snidaro, C. Micheloni, and C. Chiavedale. Video security for ambient intelligence. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 35:133 – 144, 2005. [cited at p. 17]

[SMYF10] H.T. Shandiz, S.M. Mirhassani, B. Yousefi, and M.J.R. Fatemi. Fuzzy based foreground background discrimination for probabilistic color based object tracking. *International Journal of Computer Science and Network Security*, 10:120–125, 2010. [cited at p. 26]

[SN04] R. Siegwart and I.R. Nourbakhsh. *Introduction to Autonomous Mobile Robots*. The MIT Press, 2004. [cited at p. 4]

[Sou10] Sourceforge. Opencv, intel: Open source computer vision library, http://www.intel.com/research/mrl/opencv/, 2010. [cited at p. 72, 111]

[SPB$^+$03] F. Solina, P. Peer, B. Batagelj, J. Kovac, and S. Juvan. Color-based face detection in the "15 seconds of fame" art installation. In *Mirage 2003, Conference on Computer Vision / Computer Graphics Collaboration for Model-based Imaging, Rendering, image Analysis and Graphical special Effects*, pages 38–47, 2003. [cited at p. 116]

[Spe13] IEEE Spectrum. Robot sensors and actuators, http://spectrum.ieee.org/robotics/robotics-hardwarel, 2013. [cited at p. 4, 7]

[SRT99] J. Schulte, C. Rosenberg, and S. Thrun. Spontaneous short-term interaction with mobile robots in public places. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 658–663, 1999. [cited at p. 20]

[SSA04] L. Sigal, S. Sclaroff, and V. Athitsos. Skin color-based video segmentation under time-varying illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:862–877, 2004. [cited at p. xxi, 168]

[SW96] A. Saffiotti and L.P. Wesley. Perception-based self-localization using fuzzy locations. In M. van Lambalgen L. Dorst and F. Voorbraak, editors, *Reasoning with Uncertainty in Robotics.LNAI*, pages 368–385, Berlin, DE, 1996. Springer-Verlag. [cited at p. 11]

[Tor02] V. Torra. A review of the construction of hierarchical fuzzy systems. *International Journal of Intelligen Systems*, 17:531–543, 2002. [cited at p. 29, 62, 81]

[Tor11] Jose R .A. Torreao. *Advances in Stereo Vision.* InTech, 2011. [cited at p. 6]

[UPSP10] T. Uhm, H. Park, D. Seo, and J.-Il Park. Human-of-interest tracking by integrating two heterogeneous vision sensors. In *Proceedings of the 2010 IEEE Virtual Reality Conference*, VR '10, pages 309–310, 2010. [cited at p. 27]

[VGP05] J. Vermaak, S.J. Godsill, and P. Perez. Monte carlo filtering for multi-target tracking and data association. *IEEE Transactions on Aerospace and Electronic Systems*, 41:309–332, 2005. [cited at p. 26]

[VGRC12] D. Viejo, J. Garcia-Rodriguez, and M. Cazorla. A study of a soft computing based method for 3d scenario reconstruction. *Appl. Soft Comput.*, 12(10):3158–3164, October 2012. [cited at p. 18]

[VJ01] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 511–518, 2001. [cited at p. 72, 111]

[WDOH01] I. Werry, K. Dautenhahn, B. Ogden, and W. Harwin. Can social interaction skills be taught by a social agent? the role of a robotic mediator in autism therapy. In *CT '01: Proceedings of the 4th International Conference on Cognitive Technology*, pages 57–74. Springer-Verlag, 2001. [cited at p. 15]

[WH99] Y. Wu and T. S. Huang. Vision-based gesture recognition: A review. *Lecture Notes in Computer Science*, 1739:103–115, 1999. [cited at p. 18]

[Wil97] J. Wiley. Movile robot navigation using artificial landmarks. *Journal of Robotc Systems*, 14(2):93–106, 1997. [cited at p. 7]

[WJ95] M. Wooldridge and N. R. Jennings. Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*, 10:115–152, 1995. [cited at p. 10]

[WZ08] D. Wan and J. Zhou. Stereo vision using two ptz cameras. *Computer Vision and Image Understanding*, 112(2):184 – 194, 2008. [cited at p. 23]

[XLC08] T. Xiang, M.K.H. Leung, and S.Y. Cho. Expression recognition using fuzzy spatio-temporal modeling. *Pattern Recognition*, 41(1):204–216, 2008. [cited at p. 29]

[Yan08] W. Yang. *Autonomous Robots Research Advances*. Nova Science Pub Incorporated, 2008. [cited at p. 1]

[YF94] R.R. Yager and D.P. Filev. *Essentials of Fuzzy Modeling and Control*. John Wiley & Sons, Inc, 1994. [cited at p. 25, 39, 180]

[YJJMMT07] K. Young-Joong, W. Jung-Min, and L. Myo-Taeg. Fuzzy adaptive particle filter for localization of a mobile robot. In *Proceedings of the 11th international conference, KES 2007 and XVII Italian workshop on neural networks conference on Knowledge-based intelligent information and engineering systems (2007)*, pages 41–48, 2007. [cited at p. 26]

[YKA02] M. H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: a survey. In *IEEE Trans. on Pattern Analysis and Machine Intelligence 24*, pages 34–58, 2002. [cited at p. 62, 72]

[YN12] L. Yang and N. Noguchi. Human detection for a robot tractor using omni-directional stereo vision. *Computers and Electronics in Agriculture*, 89(0):116 – 125, 2012. [cited at p. 22]

[YpB10] Wang Yan-ping and Wu Bing. Robot path planning based on modified genetic algorithm. In *Future Computer and Communication (ICFCC), 2010 2nd International Conference on*, volume 3, pages V3–725–V3–728, 2010. [cited at p. 21]

[Zad65] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965. [cited at p. 42]

[Zad75] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning-i. *Information Sciences*, 8(3):199 – 249, 1975. [cited at p. 39, 180]

[Zad99] L. A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets Syst.*, 100:9–34, April 1999. [cited at p. 39, 180]

[ZB09] W. Zheng and S.M. Bhandakar. Face detection and tracking using a boosted adaptive particle filter. *Journal of Visual Communication and Image Representation*, 2:9–27, 2009. [cited at p. 26]

[Zha12] Z. Zhang. Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, 19(2):4–10, 2012. [cited at p. 18]

[Zic12] Low-cost fpga stereo vision system for real time disparity maps calculation. *Microprocessors and Microsystems*, 36(4):281 – 288, 2012. [cited at p. 22]

[Zla01] J. Zlatev. The epigenesis of meaning in human beings, and possibly in robots. *Minds Mach.*, 11(2):155–195, 2001. [cited at p. 15]

[ZZ08] C. Zhenjiang and L. Zongli. Fuzzy particle filter used for tracking of leukocytes. In *Proceedings of the 2008 International Symposium on Intelligent Information Technology Application Workshops (2008)*, pages 562–565, 2008. [cited at p. 26]

Appendices

Appendix A

Publications

The work presented in this Thesis is original work undertaken between 2007 and 2012 at the University of Granada, Spain. Portions of this work have been published elsewhere.

International Journals (ISI)

- Paúl, R. and Aguirre, E. and García-Silvente, M. and Muñoz Salinas, R., "A New Fuzzy Based Algorithm for Solving Stereo Vagueness in Detecting and Tracking People". *International Journal of Approximate Reasoning*, Volume 53, Issue 4, pp. 693-708. 2012.
  DOI: `http://dx.doi.org/10.1016/j.ijar.2011.11.003`

- Muñoz Salinas, R. and Aguirre, E. and García-Silvente, M. and Paúl, R., "A New Person Tracking Method for Human-Robot Interaction Intended for Mobile Devices". *MICAI 2007: Advances in Artificial Intelligence Lecture Notes in Computer Science*, Volume 4827, pp. 747-757. 2007.
  DOI: `http://dx.doi.org/10.1007/978-3-540-76631-5`

International Journals (other citation indexes)

- Aguirre, E. and García-Silvente, M. and Paúl, R. and Muñoz Salinas, R., "A Fuzzy System for Detection of Interaction Demanding and Nodding Assent Based on Stereo Vision". *Journal of Physical Agents*, Volume 1, pp. 15-25. 2007.

Awaiting Acceptance from International Journals (ISI)

- Paúl, R. and Aguirre, E. and García-Silvente, M. and Muñoz Salinas, R., “A New Fuzzy Stereo-Vision System for Measuring Certain Human Responses”. *To be submitted to the International Journal of Human-Computer Studies.*

International Conferences Proceedings

- Paúl, R. and Aguirre, E. and García-Silvente, M. and Muñoz Salinas, R., ”Using Stereo Vision and Fuzzy Systems for Detecting and Tracking People”. In Proc. of the *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pp. 582-591, Vol.81, Dortmund, Germany, June 2010

- Aguirre, E. and García-Silvente, M. and Muñoz Salinas, R. and Paúl, R, ”A Fuzzy System for Interest Visual Detection Based on Support Vector Machines”. In Proc. of the *Fourth International Conference on Informatics in Control, Automation and Robotics, Robotics and Automation*, pp. 181-190, Angers, France, May 2007

National Conferences Proceedings

- Aguirre, E. and García-Silvente, M. and Muñoz Salinas, R. and Paúl, R, “A Fuzzy System for Detection of Interaction Demanding and Nodding Assent Based on Stereo Vision”. *In Proc. of the VIII Workshop on Physical Agents (WAF 2007)*, pp. 57-64. - within the II Congreso Español de Informatica taking place in Zaragoza, Spain, in 2007

Theses

- **Diploma of Advanced Studies**
  Title: "Soft Computing Techniques Applied to Visual Interest Detection in Human-Robot Interaction"; *Organization*: University of Granada, Spain; *Supervisor*: Prof. Eugenio Aguirre Molina; *Period*: 2006-2007; *Presentation Date*: July, 2007

Appendix B

Resumen en Español

Resumen

El objetivo principal del trabajo presentado en esta tesis es el desarrollo de técnicas visuales que resulten útiles para el establecimiento de una interacción natural entre seres humanos y robots. En este contexto, “natural” significa que es similar a las interacciones existentes entre los humanos. En este sentido, nuestros esfuerzos se han centrado en hacer posible que un robot, equipado con una cámara estéreo, sea capaz de analizar y estudiar el comportamiento de las personas que se encuentran a su alrededor.

La motivación subyacente a este objetivo es proporcionar a los robots la posibilidad de comportarse como lo haría un ser humano, eligiendo entre diferentes acciones de la misma manera como lo haría una persona. Esto pasa por ejecutar distintas tareas tales como: ser capaz de detectar y seguir personas en su entorno, detectar cual o cuales de entre estas personas están interesadas en las acciones propuestas por el robot, y además están respondiendo a esas mismas acciones. Por otra parte, los robots pueden utilizar sus recursos de una manera más apropiada y mejorar sus métodos de comunicación alcanzando un comportamiento más cercano al comportamiento humano.

Para alcanzar este tipo de Interacción Humano-Robot consideramos diferentes técnicas. Estas técnicas contribuyen a resolver varios problemas existentes en esta área. En particular, las técnicas de ”Soft Computing” son utilizadas para tratar la incertidumbre e imprecisión, así como para representar las variables y las reglas de una manera más comprensible por el ser humano. Son utilizadas también diferentes técnicas de análisis de imágenes para extraer la información relevante del entorno del robot. Todas estas técnicas permiten una mejora en la socialización de los robots.

El objetivo de este trabajo puede dividirse en dos. El primero es la detección y el seguimiento de las personas que se encuentran en el entorno del robot. El segundo es la detección del interés de cada persona en interaccionar con el robot, la detección de la demanda de atención al robot y la detección de la respuesta a sus acciones. Esto se realiza en base al análisis de algunos de los elementos que caracterizan una situación de interacción típica entre humanos tales como: la distancia entre los diferentes interlocutores, la orientación de la cabeza, el movimiento de brazos, el movimiento de concordancia y discordancia entre la cabeza y la expresión de la boca (sonrisa).

Para alcanzar el primer objetivo se consideran dos métodos: el primero basado en un enfoque probabilístico y el segundo basado en un enfoque "posibilístico". El método probabilístico muestra un nuevo enfoque para el seguimiento de personas que combina profundidad, color e información de gradiente y está basado en visión estéreo. El grado de confianza asignado a la información de profundidad en el proceso de seguimiento varía de acuerdo con la cantidad de información estéreo disponible en el mapa de disparidad. Se ha definido una nueva medida de confianza para alcanzar este objetivo y el seguimiento se hace utilizando filtros de partículas. El segundo método, basado en un enfoque "posibilístico", se utiliza para añadir más información basada en conocimiento experto que se usa a la hora de evaluar las partículas. Este enfoque tiene las restricciones derivadas de las condiciones de un modelo probabilístico. En este caso se utiliza la lógica difusa para manejar la información estéreo y así poder detectar y seguir a las nuevas personas. Más concretamente, en la fase de detección de personas, se utilizan dos sistemas difusos para filtrar los falsos positivos del detector de caras. A continuación, en la fase de seguimiento, se propone un nuevo Filtro de Partículas basado en Lógica Difusa para fusionar la información estéreo y la información de color, asignando diferentes niveles de confianza a cada una de estas fuentes de información. De esta manera, el sistema es capaz de seguir a las personas, en la imagen de referencia de la cámara, aún cuando una de las fuentes de información utilizada (estéreo o color) sea confusa o imprecisa.

Considerando que un robot es un sistema inteligente, la detección de determinadas situaciones de interacción es una habilidad que resulta interesante. Por consiguiente, para alcanzar el segundo objetivo, se presenta un método basado en diferentes características, como el ángulo y la distancia entre las personas y el robot, así como la dirección de la cabeza de cada persona. La estimación de la dirección de la cabeza en tiempo real se hace utilizando una técnica basada en ”Support Vector Machines” mientras que se utiliza un sistema difuso para calcular el valor de interés final a partir de las tres variables que se acaban de mencionar. Siempre que el grado de interés alcanza un valor alto, la persona se analiza con más en detalle para detectar su posición y un determinado tipo de movimiento de sus brazos y cabeza (concordancia y discordancia). Esta información se gestiona por otro sistema difuso que debe calcular si la persona está llamando la atención del robot o si está diciendo SI / NO con su cabeza. En el último trabajo presentado en esta tesis, algunas de estas fuentes de información se usan de forma conjunta con una técnica de detección de sonrisa, para construir un sistema basado también en lógica difusa, que tiene la capacidad de medir ciertos tipos de respuesta humana. Como la fiabilidad de la información visual captada por la cámara estéreo depende bastante de la distancia de cada persona con respecto a la cámara, las diferentes características visuales se priorizan de acuerdo con la distancia de la persona al robot. La respuesta humana se calcula a partir de un sistema difuso jerárquico que es capaz de tratar la incertidumbre y la imprecisión existentes en dichas medidas, según la distancia a la que se encuentra la persona con respecto al robot. Esta medición de la respuesta humana se utiliza para detectar la persona o las personas que están respondiendo mejor a la interacción social propuesta por el robot. Dicha medición puede servir también para mejorar y ajustar las habilidades de interacción social del robot en el futuro.

Preámbulo

Como se ha mencionado en el resumen, para que un sistema artificial actúe de manera “natural”, hay ciertos comportamientos que se deben llevar a cabo de una forma similar a como los realizan los seres humanos. En este trabajo, se asume que el robot se encuentra completamente inmóvil durante el proceso de interacción. La misma condición se aplica a la cámara estéreo. Esta condición puede ser considerada aceptable ya que típicamente, cuando una persona está analizando el comportamiento de otra e intentando entender como su interlocutor reacciona con respecto a sus acciones, como en el caso de nuestro robot, dicha persona se queda normalmente quieta y enfocando su atención en su(s) interlocutor(es). Obviamente, esa persona puede empezar a moverse durante la interacción pero no es el objetivo de este trabajo permitir que el robot pueda moverse mientras está analizando a sus interlocutores. Sin embargo, aunque el robot se encuentre parado, sus interlocutores sí que se pueden mover libremente en su entorno.

El primer problema que se toma en consideración en este trabajo es la capacidad de detectar y seguir correctamente a las personas que se encuentran en el entorno del robot. La detección y el seguimiento de personas se pueden realizar de diversas maneras y utilizando diferentes tipos de hardware. Cuando se utilizan técnicas de visión artificial, el sistema debe analizar las imágenes y buscar los elementos que proporcionan información relevante para la detección de los objetivos. Esos elementos pueden ser, por ejemplo, características morfológicas del cuerpo humano ([HM03]) y modelos dinámicos del color de piel ([SSA04]).

Existen varios métodos empleados en el seguimiento de personas que se fundamentan en la información de color de la ropa de las personas. Generalmente, el primer paso es crear el modelo de color de la persona que se ha detectado y que va a ser seguida. A continuación, en la secuencia siguiente de imágenes, la posición y el tamaño de la región de la imagen que mejor coincide con el modelo de color de la persona, son considerados como la nueva posición y tamaño de la persona que está siendo seguida. A esta técnica se llama seguimiento adaptativo y es especialmente indicada para el seguimiento de ”non-rigid targets”, con respecto a los cuales no hay un modelo explícito o bien cuando la estimación del background no es posible.

Como la mayoría de estas técnicas se fundamenta únicamente en la información de color, se constatan varios inconvenientes. El más importante es la confusión entre dos o más áreas de la imagen que tienen la misma distribución de color y siempre que se encuentren cerca una de la otra. Como no hay otra información que permita distinguir entre ellas, esto puede llevar al sistema a confundir los distintos objetivos. Esta confusión puede también producirse con respecto al background, siempre que el algoritmo de seguimiento no sepa qué partes de la imagen forman parte del background o del foreground. Así, en el caso de que el background, o una de sus partes, tengan una distribución de color similar a la persona que está siendo seguida, estos sistemas puede perder el objetivo. Finalmente, existe también la posibilidad de que una subregión de la persona sea considerada como la totalidad de la misma. Esto puede llevar a que la determinación del tamaño de la persona sea imprecisa e incorrecta y esto genera posteriormente otros problemas.

Algunos autores han propuesto el uso de la tecnología estéreo para resolver esta cuestión. Esta tecnología ha sido bastante estudiada en los últimos años y se emplea cada vez más en distintas aplicaciones ([BBH03]). Con el desarrollo de tecnologías estéreo bien consolidadas, y la disponibilidad de hardware comercial capaz de solucionar los problemas inherentes a la visión estereoscópica, esta técnica se ha convertido en una herramienta importante a la hora de desarrollar aplicaciones basadas en visión por ordenador, como los algoritmos de seguimiento. Estos algoritmos pueden aprovechar la información de distancia a la que se encuentra cada pixel para solucionar algunos de los problemas de los algoritmos tradicionales de visión. En primer lugar, la posibilidad de conocer la distancia a la que se encuentra cada persona con respecto a la cámara puede ser una ayuda importante. En segundo lugar, la información de distancia es menos sensible a los cambios de iluminación con respecto a la información dada por una sola cámara.

A partir del momento en que el problema de detección y seguimiento de personas se encuentra resuelto, se deben analizar otros problemas con vista al desarrollo de un robot con capacidades sociales. Existen distintos enfoques que pueden contribuir al progreso de la difícil tarea de construir un robot social ([FND03]). En ese trabajo, el rango de problemas que se tienen que considerar puede variar desde el diseño del robot hasta su aceptación por la sociedad, desde la detección de emociones hasta la expresión de sus propias emociones, desde la posibilidad de simular una personalidad hasta la imitación de otras personalidades. En todos los casos, un robot social debe estar preparado para captar e interpretar los elementos de comunicación empleados por sus interlocutores y así poder completar sus tareas con éxito. Además, lo deberá hacer de manera natural, utilizando características visuales naturales.

Aunque diferentes autores hayan contribuido con distintos trabajos en este campo ([FND03]) (ver Sección 1.3), existe aún un rango amplio de cuestiones por explorar que permitirán mejorar la interacción entre los robots sociales y los humanos. En este trabajo, hemos centrado nuestros esfuerzos en reconocer cuando y por cuanto tiempo una persona está interesada en establecer una interacción. Así como en conocer cual es el nivel de respuesta de esas personas a las actividades propuestas por un robot social. Para resolver estas tareas se pueden tener en cuenta diferentes tipos de señales de comunicación expresadas por los seres humanos (verbales y no verbales). Algunos autores ([BFJ$^+$05a]) utilizan la localización de sonido y el reconocimiento de voz, combinado con la percepción visual para detectar la(s) persona(s) más interesadas en una interacción. En otros casos, se analizan expresiones faciales ([SKKB01]) y gestos ([GNS$^+$02]). En este trabajo, se dedica un interés especial al análisis de algunas situaciones típicas de interacción, las cuales se pueden integrar en el futuro en sistemas más complejos.

Este trabajo se basa en el sentido humano más importante a la hora de interaccionar: la visión. Además, el uso de la visión hace innecesario el uso de dispositivos específicos adicionales, como ocurre con otros tipos de sensores. Esto hace que la forma de percibir el mundo para el robot sea análoga a la humana. La información visual tiene características de imprecisión que provocan que sea necesario utilizar herramientas que manejen incertidumbre. Por esta razón, usamos lógica difusa como modelo para el tratamiento de la información. En especial, los diferentes elementos visuales definidos en este trabajo pueden indicar, de manera natural, si una persona está interesada, colaborando y/o respondiendo a una interacción. Los elementos visuales que se han tenido en cuenta para inferir el nivel de respuesta humana y su interés son la orientación de la cabeza con respecto al robot, el movimiento de los brazos y si la persona está o no sonriendo. También pensamos que una persona que quiere interaccionar y comunicarse con otra persona intentará tener línea de visión directa con su interlocutor y evitará estar ocluida por otras personas u objetos. Adicionalmente, se presenta un método para inferir la posibilidad de que una persona esté contestando SI o NO con su cabeza. Como se puede observar, estos elementos visuales son "naturales" y son bastante utilizados cuando interaccionamos con otras personas.

Para alcanzar los objetivos propuestos se han tenido en cuenta diferentes restricciones. Por ejemplo, es casi imposible detectar con certidumbre una sonrisa (considerando la resolución del dispositivo) cuando una persona se encuentra a más de un par de metros de la cámara. Otra restricción tiene que ver con el hecho de que cuando una persona se encuentra bastante cerca de la cámara sus brazos están fuera del campo de visión y, consecuentemente, nos es posible detectar el movimiento de sus brazos. De esta manera, es fácil entender que la distancia juega un papel muy importante con respecto a la detección de los diferentes elementos visuales propuestos. Así, en la propuesta presentada en esta tesis, la respuesta humana es calculada por medio de un sistema difuso jerarquizado que es capaz de tratar la incertidumbre y la imprecisión de las diferentes percepciones visuales en función de la distancia de la persona respecto a la cámara. Al medir esta respuesta, el robot tiene la posibilidad de interaccionar de una forma más natural y mejorar así la actividad que está proponiendo a sus interlocutores según la retroalimentación recibida de los participantes.

Contribuciones Principales

El trabajo descrito en esta tesis creemos que ha contribuido con distintos avances en el área de Interacción entre Robots y Humanos. No solamente se han probado diferentes metodologías, sino que se han propuesto nuevas ideas y nuevos enfoques que pueden ayudar al desarrollo de este campo. Entre estas contribuciones las más relevantes son:

1. Un nuevo método que permite seguir personas que está basado en un filtro de partículas que integra profundidad, color e información de gradiente para alcanzar un seguimiento más robusto. Este método incluye una medida de certeza que indica la confianza que se tiene sobre la información de profundidad y permite de esta forma manejar problemas provocados por las oclusiones y/o falta de textura. Este trabajo se ha publicado en [MSAGSP07].

2. Un sistema capaz de detectar y seguir varias personas al mismo tiempo utilizando un nuevo enfoque basado en color, visión estéreo y lógica difusa. Inicialmente, en la fase de detección de personas, se utilizan dos sistemas difusos que permiten filtrar los falsos positivos provocados por el detector de caras. A continuación, en la fase de seguimiento, se usa un nuevo filtro de partículas basado en lógica difusa con el objetivo de fusionar la información estéreo y de color, asignando diferentes niveles de confianza a cada una de estas fuentes de información. La información de profundidad y de oclusión se utiliza para crear estos niveles de confianza. De esta manera, el sistema es capaz de seguir a varias personas en la imagen de referencia de la cámara a pesar de que la información correspondiente al color o a la profundidad sea confusa o imprecisa.

   Este trabajo se ha publicado en [PAGSMS12].

3. Un nuevo sistema difuso que permite la detección visual de posibles demandas de interacción, además de la detección de gestos de concordancia y discordancia realizados con la cabeza. El nivel de interés de la persona para interaccionar con el robot se calcula analizando su posición, la orientación de su cabeza y el movimiento de sus brazos. La estimación de la orientación de la cabeza se realiza en tiempo real a través de un enfoque basado en Máquinas de soporte vectorial (SVMs). Este trabajo se ha publicado en [AGSG$^+$07].

4. Un sistema capaz de medir la respuesta humana de personas que se encuentran en el entorno de un robot social utilizando lógica difusa y visión estéreo. Para alcanzar este objetivo, el sistema analiza diferentes pistas visuales que los humanos utilizan de forma ”natural” y que proporcionan al robot una retroalimentación con respecto a la actividad que éste se encuentra proponiendo. La respuesta humana se calcula a través de un sistema jerárquico difuso que es capaz de tratar la incertidumbre y la imprecisión presentes en la información proveniente de los sensores, y que dependerá de la distancia a la cual se encuentre la persona del robot. Este trabajo está sometido a la revista International Journal of Human Computer Studies (IJHCS).

Estructura de la tesis

Esta tesis está organizada en varios capítulos. El primero corresponde a la Introducción. En este Capítulo se presenta también una revisión del estado del arte sobre los diferentes campos de la Interacción entre Humanos y Robots, así como una corta descripción de otros trabajos que tratan temas directamente relacionados con esta tesis. En el Capítulo 2, se describe la configuración del sistema que se ha utilizado para el desarrollo, implementación y experimentación del trabajo realizado. También se presentan algunas de las técnicas utilizadas como la visión estéreo y la modelización del color, el análisis de componentes principales (PCA), las Máquinas de soporte vectorial (SVM) y la lógica difusa (FL).

En los Capítulos 3, 4 y 5 se presentan las contribuciones más importantes de esta tesis. En el Capítulo 3, se muestran dos métodos diferentes de detección y seguimiento de personas: uno probabilístico y el otro posibilístico. En el Capítulo 4 presentamos un método para la detección del interés y la demanda de atención. Finalmente, en el Capítulo 5 se presenta un método para la detección de la respuesta humana.

En el Capítulo 6 se muestran las Conclusiones, algunas consideraciones finales y los trabajos futuros.

Descripción del Sistema y de las Técnicas Empleadas

En este capítulo, se describe el hardware de nuestro sistema. A continuación, se muestran los elementos básicos para el desarrollo de los métodos propuestos: la base de la visión estéreo y el modelado de color. En la tercera sección se explican dos técnicas usadas en aprendizaje automático (Machine Learning (ML)) que son el análisis de componentes principales (Principal Component Analysis (PCA)) y las máquinas de soporte vectorial (SVM). Estas técnicas se emplean en diferentes ocasiones a lo largo de esta tesis. En la última sección de este capítulo se presentan algunos fundamentos de lógica difusa, como método de representación de la imprecisión e incertidumbre y como herramienta de ”razonamiento”.

Descripción del Hardware

El hardware utilizado en esta tesis está compuesto por un robot móvil “PeopleBot” [Rob], un sistema estereoscópico con una cámara binocular [Res05] y un portátil encargado de procesar la información visual. La cámara y el portátil se encuentran montados en la parte superior del robot como se puede observar en la Figura 2.1. El sistema estereoscópico utilizado no solo permite la extracción de color sino también la información de profundidad. En nuestros experimentos, se han grabado varias secuencias con una resolución de 320x240 píxeles a una tasa de 15 frames por segundo. Al principio, se utilizó un portátil con un procesador Intel Pentium IV a 3.2 Ghz. En los últimos trabajos, se ha cambiado a otro portátil con un Intel i5 a 2.67 Ghz.

El sensor principal utilizado en los diferentes trabajos, la cámara estéreo, es capaz de captar dos imágenes ligeramente distintas (par estéreo calibrado). Estas imágenes se utilizan para calcular la imagen de disparidad entre las dos imágenes captadas.

Aunque la funcionalidad del movimiento del robot no se utiliza directamente en los diferentes trabajos de esta tesis, en un futuro se podrán integrar los algoritmos presentados aquí con otros trabajos desarrollados en nuestro grupo de investigación. La altura del robot es similar a la altura media de un niño de 8 a 10 años. Pensamos que esta característica es una ventaja que favorece la interacción entre un robot y las personas. De hecho, las personas se sentirán más confortables al interaccionar con un robot con una altura parecida a la de un ser humano que, además, tiene su sistema de visión ubicado en su parte superior tal y como un ser humano. Esta idea ha sido una de las condiciones que hemos tenido en cuenta en los trabajos desarrollados. Así respetamos uno de los objetivos del grupo de investigación que es el de simular las mismas condiciones que se pueden dar en una interacción entre seres humanos.

El sistema propuesto en esta tesis está diseñado para utilizarse en diferentes actividades en las cuales pueden participar hasta cuatro personas al mismo tiempo. Dichas personas pueden moverse e interaccionar entre sí y/o con el robot, dentro de un rango de distancias que puede variar entre 0.5 y 5 metros (estas limitaciones provienen de la resolución de la cámara, del campo de visión y del tiempo de procesamiento adecuado para las aplicaciones en tiempo real).

Visión por Computador

En esta sección se describen las dos técnicas principales de visión artificial que se utilizan en los diferentes trabajos de esta tesis. En primer lugar se presenta la visión estéreo y a continuación el modelo de color.

Visión Estéreo

En esta sección se presentan algunos fundamentos de visión estéreo[BBH03]. Es importante advertir que no es nuestro objetivo desarrollar o presentar un nuevo algoritmo de emparejamiento estéreo y que se utiliza el software proporcionado por el fabricante de la cámara[Res10] para obtener la información de profundidad. El software de la cámara resuelve cuestiones como la distorsión de la óptica en el momento de realizar el cálculo estéreo. También, el software aporta un núcleo optimizado para el cálculo eficiente del estéreo basado en la Suma de las Diferencias Absolutas. Este método es conocido por su rapidez, simplicidad y robustez y es capaz de generar imágenes densas de disparidad.

Un sistema de visión estéreo básico está compuesto por un par de cámaras colocadas de forma paralela y cuyos centros ópticos (O_l y O_r) están separados por una distancia b. Empecemos por asumir, para simplificar la explicación, que ambas cámaras tienen características ópticas equivalentes y que los planos de visión son coplanarios (como se muestra en la figura 2.2). Un sistema de visión estéreo es capaz de producir dos imágenes (I_l y I_r) en el mismo instante. Las dos cámaras se encuentran calibradas y las imágenes capturadas son rectificadas para eliminar las deformaciones causadas por la distorsión de las ópticas. Normalmente, se define como sistema de referencia el centro de una de las cámaras. En nuestro caso, será el centro de la imagen derecha (al cual llamamos imagen de referencia de la cámara).

Un punto en el espacio $P = (X, Y, Z)$ se proyecta en dos puntos distintos ($p = (x, y)$ y $p' = (x', y')$) en la misma línea epipolar en cada imagen rectificada. A la distancia entre las proyecciones de ese punto en cada uno de los planos de las cámaras se llama disparidad y el conjunto de todas las disparidades entre los puntos de las dos imágenes es lo que constituye el mapa de disparidad. Las disparidades solo se pueden calcular para los puntos aparecen en ambas imágenes y además hay variedad en la textura. Consecuentemente, es difícil calcularlo cuando hay oclusiones. Los puntos en los que no se puede calcular la disparidad se denominan puntos sin correlación.

Al ser conocidos los parámetros intrínsecos del sistema de visión estéreo, como por ejemplo la distancia focal (para nuestro sistema es 6mm), es posible reconstruir la estructura tridimensional que corresponde al mapa de disparidad.

En la figura 2.3 se puede observar un ejemplo de una escena captada por un sistema de visión estéreo. En la figura 2.3a tenemos la imagen izquierda I_l y en la figura 2.3b la imagen derecha I_r (que ha sido definida como imagen de referencia). En la figura 2.3c tenemos la imagen de distancia I_z. En esta imagen, los píxeles con un tono más claro indican valores más bajos de Z (y mayor disparidad) mientras que los píxeles más oscuros representan distancias mayores (y menor disparidad). Los píxeles negros representan los puntos sin correlación. El mapa de disparidad para este frame sería una imagen similar a la de la figura 2.3c. También es importante tener en cuenta que la información de distancia obtenida de un par estéreo puede estar afectada por errores típicos del estéreo producidos en las fases de calibración, cuantización y correlación[MMN89][RA90]. Por tanto, los algoritmos que utilizan información estéreo deben estar preparados para tratar de forma adecuada estos errores.

La información de distancia (estéreo) se emplea para mejorar el algoritmo de seguimiento, y éste realmente se lleva a cabo en la imagen de referencia, en un dominio de dos dimensiones. Se considera como la posición de la persona el centro de su cara, que ha sido previamente detectado por un detector de caras en la imagen de referencia. La posición de la persona se escribe como (x_p, y_p) y se corresponde con un pixel concreto de la imagen de referencia.

Modelo de color

La utilización de la información de color en el seguimiento de objetos es un problema que ha sido bastante estudiado[Bir98][CR00][GK04][NKMG03].

Los métodos más frecuentemente utilizados consisten en usar un histograma para representar un modelo de color $\hat{q}$. Ya que las componentes HS del espacio de color HSV ([FvD82]) son relativamente invariables a cambios de iluminación, se ha convertido en un enfoque popular en este dominio. Un histograma de color $\hat{q}$ está constituido por $n_h n_s$ columnas para la tonalidad y la saturación. Sin embargo, la información cromática no puede ser considerada fiable cuando el valor de este componente es demasiado bajo o demasiado alto. Así, estos píxeles se eliminan a la hora de describir el modelo. Debido al hecho de que estos píxeles pueden contener información importante, el histograma también tiene en cuenta n_v columnas para capturar la información de luminosidad. Por tanto, el histograma final estará compuesto por $m = n_h n_s + n_v$ columnas.

Se utiliza una región elíptica de la imagen para crear un modelo de color en la cual h_x y h_y representan los ejes horizontal y vertical respectivamente[Bir98][CR00][NKMG03]. Siendo p_c el centro de la elipse y $\{p_j\}_{j=1,...,n}$ los píxeles interiores de la misma. Se define también la función $b : \Re^2 \rightarrow 1, ..., m$ que asocia al pixel p_j el índice $b(p_j)$ de la columna del histograma correspondiente al color u de ese pixel. Se puede entonces calcular la distribución de densidad de color $\hat{q}$ para cada región elíptica como:

$$\hat{q}(u) = \frac{1}{n} \sum_{j=1}^{n} k[b(p_j) - u], \tag{B.1}$$

Donde el parámetro k es la función delta Kronecker. Se debe tener en cuenta que el histograma final se normaliza, o sea, $\sum_{u=1}^{m} \hat{q}(u) = 1$.

Después de calcular el modelo de color $\hat{q}$, es posible compararlo con otro modelo de color $\hat{q}'$, utilizando una medida de similitud. Habitualmente se utiliza el coeficiente de Bhattacharyya[ATR97][Kai67]. En el caso de una distribución discreta, podemos expresarlo como se indica en la ecuación 2.2. El resultado indica la similitud entre dos modelos de color en un rango que varía entre $[0, 1]$ y donde 1 significa que los dos modelos son idénticos y 0 indica que son completamente diferentes. Una característica importante de ρ es que dos modelos de color, $\hat{q}$ y $\hat{q}'$ se pueden comparar aunque hayan sido creados a partir de regiones con diferentes tamaños. En la parte izquierda de la figura 2.4 podemos encontrar un ejemplo de un frame extraído de un video y en la derecha una tabla en la que se comparan diferentes regiones de interés y donde se indica el coeficiente de Bhattacharyya en cada caso.

$$\rho(\hat{q}, \hat{q}') = \sum_{u=1}^{m} \sqrt{\hat{q}(u)\hat{q}'(u)}. \tag{B.2}$$

Aprendizaje Automático

En esta sección se presentan dos técnicas que se utilizan en el ámbito del aprendizaje automático y que se usan más adelante. Dichas técnicas son: Análisis de componentes principales (PCA) y Máquinas de soporte vectorial (SVM).

Análisis de componentes principales

El Análisis de componentes principales (PCA)[HD89] es una técnica que se usa para reducir la dimensionalidad de los datos pero conservando la esencia fundamental de los mismos. Cuando se utiliza PCA, se transforma una imagen en sus componentes principales, o sea, las componentes que contienen los aspectos "más importantes" de la información. PCA permite la identificación de patrones en los datos, de manera que permite resaltar sus similitudes y sus diferencias. PCA también se utiliza para comprimir información.

Una proyección PCA representa un conjunto de datos en términos de vectores propios ortonormales de la matriz de covarianza de los datos. La matriz de covarianza indica la correlación entre las diferentes variables en un conjunto de datos. PCA detecta los vectores propios ortonormales de la matriz de covarianza como base para el espacio de características. Los vectores propios pueden ser vistos como una "base natural" para un determinado conjunto de datos multi-dimensional. Valores propios mayores en la matriz de covarianza indican una correlación menor entre las características del conjunto de datos. Las proyecciones del PCA buscan las variables no relacionadas.

A todos los conjuntos de datos se les puede extraer sus componentes principales pero el PCA funciona mejor si los datos siguen una distribución gaussiana. Cuando tenemos una gran cantidad de datos, el teorema del límite central nos permite asumir distribuciones gaussianas.

Empecemos por calcular la varianza de una variable x como:

$$\sigma^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{X})^2}{n}$$

Luego, se calcula la varianza de dos variables x y y como:

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(x_i - \bar{X})(y_i - \bar{Y})}{n}$$

A partir de la covarianza, podemos observar como evolucionan dos variables una en función de la otra:

- En el caso de que la covarianza entre dos variables sea positiva, si una variable aumenta, la otra aumentará también.
- En el caso de que la varianza entre las dos variables sea negativa, si una variable aumenta, la otra disminuye.
- En el caso de que la varianza entre las dos variables es cero, entonces las dos variables son completamente independientes.

Para un conjunto de variables $< X_1, ..., X_n >$, (por ejemplo, las características de un conjunto de datos) es posible construir la matriz que representa la varianza entre cada par de variables X_i‘ y X_j donde i y j son los índices del vector de características.

$$cov(X) = \begin{bmatrix} var(X_1) & cov(X_1, X_2) & \cdots & cov(X_1, X_n) \\ cov(X_2, X_1) & var(X_2) & \cdots & cov(X_2, X_n) \\ \vdots & \vdots & \ddots & \vdots \\ cov(X_n, X_1) & cov(X_n, X_2) & \cdots & var(X_n) \end{bmatrix}$$

Para esta matriz, es posible observar que en la diagonal se representa simplemente la varianza de una variable individual y que la matriz es simétrica, lo que significa que $cov(X_i, X_j) = cov(X_j, X_i)$.

Antes de utilizar el concepto de covarianza en PCA, hay un primer paso que consiste en sustraer las medias $\bar{X}_i$ de cada x_i para que cada $\bar{X}_i$ tenga una media igual a cero. Al sustraer la media es posible reescribir la matriz de covarianza como el siguiente producto:

$$\mathbf{\Sigma = \frac{1}{n} X X^T}$$

Luego, a través de la teoría de descomposición, se puede factorizar la matriz mencionada anteriormente en:

$$\mathbf{\Sigma = U \Lambda U^T}$$

donde $\mathbf{\Lambda} = diag(\lambda_1, \cdots, \lambda_n)$ es la matriz diagonal de valores propios de la matriz de covarianza ordenada del mayor al menor:

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}$$

A continuación, las componentes principales son los vectores de $\mathbf{U^T}$. $\mathbf{U^T}$ representa la proyección de la matriz de peso W y la matriz de los datos transformados S se puede obtener a partir de la matriz de datos originales X como:

$$\mathbf{S} = \mathbf{WX}$$

Si no seleccionamos los vectores propios que corresponden a los valores propios más bajos entonces cada s tendría una dimensión más baja que su x correspondiente. El hecho de descartar estos vectores propios puede ser visto como la acción de descartar ruido en los datos.

Máquinas de soporte vectorial

Las máquinas de soporte vectorial (SVM) es una técnica útil para clasificar datos. Aunque SVM se considera más fácil de usar en comparación con las redes neuronales, normalmente las personas que no están familiarizadas con este proceso no consiguen obtener resultados satisfactorios al principio. Además, hay dos ventajas principales de SVM con respecto a las redes neuronales. En primer lugar, gran parte de los diferentes tipos de redes neuronales pueden caer en mínimos locales mientras que la solución proporcionada por SVM es única y global. En segundo lugar, la complejidad computacional de SVM no depende de la dimensión de los datos de entrada, lo que no sucede con las redes neuronales. Aunque los lectores no necesiten entender la teoría subyacente de las SVM, vamos a describir de forma breve ciertos fundamentos para poder comprender mejor la herramienta utilizada. En nuestro caso hemos aplicado una biblioteca que se encuentra disponible gratuitamente en internet [CL11] y la explicación que se muestra a continuación se ha tomado de ese mismo recurso.

Una tarea de clasificación normalmente implica que antes se realice una separación entre los datos de entrenamiento y los datos de prueba. Cada ejemplo que pertenece a los datos de entrenamiento se clasifica en una clase con unos determinados “atributos” (por ejemplo, las características observadas). El objetivo de las SVM es producir un modelo (basado en los datos de entrenamiento) que sea capaz de predecir la clase a la que pertenecen los datos de prueba a partir de los atributos de esos mismos datos.

Dado un conjunto de entrenamiento de pares de instancias etiquetadas $(x_i, y_i), i = 1, ..., l$ donde $x_i \in R^n$ y $y_i \in \{1, -1\}^l$, aplicar SVM significa obtener la solución al problema de optimización siguiente ([BGV92][CV95]):

$$\begin{aligned}&\min_{\mathbf{w},b,\xi}\ \tfrac{1}{2}\mathbf{w}^T\mathbf{w} + C\textstyle\sum_{i=0}^{l}\xi_i\\ &\text{sujeto a } y_i(\mathbf{w}^T\phi(x_i)+b) \geq 1-\xi_i,\\ &\xi_i \geq 0.\end{aligned}$$

Aquí, los vectores de entrenamiento x_i se mapean a un espacio de grandes dimensiones a través de la función ϕ. La SVM se encarga de encontrar un híper plano de separación con el máximo margen en este espacio de grandes dimensiones. $C > 0$ es un parámetro de penalización del término de error. Además, a $K(x_i, x_j) \equiv \phi(x_i)^T\ \phi(x_j)$ se le llama la función núcleo. Aunque haya nuevos núcleos que están siendo propuestos por los investigadores, los cuatro siguientes son los más utilizados:

- lineal: $K(x_i, x_j) = x_i^T x_j$.
- polinomial: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$.
- función de base radial (RBF): $K(x_i, x_j) = exp(-\gamma\|x_i - x_j\|^2), \gamma > 0$.
- sigmoideo: $K(x_i, x_j) = \tanh(x_i^T x_j + r)$.

Lógica Difusa

El principal sistema de razonamiento utilizado en los diferentes trabajos de esta tesis es la lógica difusa. A continuación presentaremos algunos fundamentos de la lógica difusa, principalmente los que se han empleado en dichos trabajos[Ful10][Kae]. Los lectores más interesados en profundizar en cuestiones de lógica difusa pueden consultar referencias como [Zad75], [Zad99], [DD96] y [YF94].

La lógica difusa fue concebida por Lofti Zadeh como una manera de procesar datos permitiendo una pertenencia parcial en vez de una pertenencia binaria. Según él, las personas no necesitan información precisa y numérica como entrada y, aun así, son capaces de realizar acciones de control altamente adaptativas. Él creía que la retroalimentación de los controladores podría ser programado de una manera tal que le permitiría aceptar y procesar entradas ruidosas y/o imprecisas y, al hacerlo, podrían convertirse en sistemas más eficaces y sencillos en su implementación.

Así, podemos definir la lógica difusa como una metodología de control orientada a la resolución de problemas que puede ser implementada en un amplio rango de sistemas diferentes. Puede ser implementada a nivel del hardware, de software o de ambos. La lógica difusa proporciona una manera sencilla de llegar a una conclusión basada en información vaga, ambigua, imprecisa o ruidosa. Un enfoque de control basado en lógica difusa, imita la toma de decisiones por parte de las personas, pero de manera mucho más rápida.

Entre los beneficios de utilizar la lógica difusa, hay algunos que nos gustaría resaltar:

- Se considera una técnica robusta que no necesita información precisa y libre de ruido. Aún en el caso de un fallo en la entrada, el sistema puede seguir "trabajando" y su salida es normalmente fluida, aunque puedan existir diferentes tipos de entradas.

- Es bastante modular y fácilmente adaptable. De hecho, sus reglas pueden ser cambiadas fácilmente y afinadas para ajustar el funcionamiento del sistema. Nuevos sensores se pueden incorporar fácilmente mediante la definición de nuevas variables y reglas y/o adaptando las existentes.

- Permite la utilización de un amplio rango de sensores ya que no se encuentra limitada a solo algunos tipos de entradas de retroalimentación o una o dos salidas de control. No es necesario medir o calcular el ratio de cambio de los parámetros para que sea implementada. Es posible utilizar sensores de bajo coste o bien sensores imprecisos, logrando un sistema con complejidad y coste bajos.

- Se puede procesar un numero razonable de entradas (1-8 o más) y varias salidas (1-4 o más). Sin embargo, la complexidad de la base de reglas puede aumentar cuando se utilizan muchas entradas, así que es aconsejable el distribuir diferentes tareas a diferentes controladores (utilizando, por ejemplo, un enfoque basado en sistemas difusos jerárquicos (HFS)).

- También es posible modelar sistemas no-lineales dando lugar a la modelización de sistemas de control que normalmente serían considerados como "no reproducibles" de forma automática.

Además, Lotfi Zadeh propuso el concepto de variables lingüísticas o "difusas". Podemos verlas como palabras u objetos lingüísticos y no números. La entrada proveniente del sensor es un sustantivo (por ejemplo "temperatura", "desplazamiento", "velocidad", "flujo", "presión", etc). El error cometido puede ser visto de la misma manera. Las variables difusas son adjetivos que modifican la variable (por ejemplo "gran error positivo", "pequeño error positivo", error "cero", "pequeño error negativo", y "gran error negativo"). Para simplificar, podemos utilizar solamente las variables "positivo", "cero", y "negativo" para cada uno de los parámetros. Variables adicionales tales como "muy grande" y "muy pequeño" también podrían añadirse para extender la capacidad de respuesta a condiciones excepcionales o bastante no lineales, pero no suelen ser necesarias en un sistema básico.

La lógica difusa incorpora un enfoque sencillo, basado en reglas del tipo *IF X AND Y THEN Z* para encontrar la solución a un problema de control en vez de intentar la descripción del sistema en la forma clásica matemática. El modelo de lógica difusa se fundamenta en información empírica, como la experiencia de un operador. Por ejemplo, en el caso de control de temperatura, en vez de tratarla de la forma "SP = 500F", "T < 1000F", o "210C < TEMP < 220C", se utilizan términos como *"IF (proceso está demasiado frío) AND (proceso se está enfriando) THEN (añadir calor al proceso)"* o *"IF (proceso está demasiado caliente) AND (proceso está calentando rápidamente) THEN (enfriar proceso rápidamente)"*.

La siguiente cuestión lógica es como aplicar las reglas. Esto nos lleva al siguiente concepto, el de función de pertenencia. La función de pertenencia es una representación gráfica de la magnitud de participación de una variable difusa de cada entrada al sistema. Las reglas utilizan la función de pertenencia como factor de ponderación en el cálculo de los conjuntos difusos de salida. Cada entrada y cada salida tienen asociadas diferentes funciones de pertenencia.

Los resultados producidos por la activación de cada regla deben inferirse y luego combinarse antes de llevar a cabo el proceso de defuzificación, el cual genera una salida numérica concreta.

A continuación, el sistema debe ser ajustado para producir los mejores resultados. Esto puede hacerse cambiando los antecedentes o las conclusiones de las reglas, cambiando los centros de las funciones de pertenencia de entrada y/o salida, o añadiendo otros niveles a las funciones de entrada y/o salida como, por ejemplo, niveles de "error", "error-dot", y respuesta de salida "bajos", "medios", y "altos". Estos nuevos niveles generan nuevas reglas y funciones de pertenencia que van a sobreponerse a las funciones adyacentes formando más "rangos" de funciones y respuestas.

Conclusiones y Trabajo Futuro

Esta tesis presenta diferentes trabajos llevados a cabo en los últimos años y que están relacionados con diferentes áreas de las Ciencias de la Computación como la Inteligencia Artificial, la Interacción entre Robots y Humanos y la Visión por Computador. En particular, nuestros esfuerzos se han enfocado en la problemática de la detección y seguimiento de personas que consideramos un tema primordial y que debe ser resuelto antes de investigar en técnicas de Interacción entre Robots y Humanos. A continuación, hemos desarrollado algunas técnicas para la detección de diferentes tipos de respuesta humana. En los distintos capítulos de esta tesis ya se han mostrado algunas conclusiones parciales. En este último capítulo, el objetivo es presentar una visión en conjunto de todos los trabajos así como las contribuciones globales de esta tesis. Como se comentó en el Resumen, son dos las problemáticas abordadas con más detalle en esta memoria. La primera relacionada con la detección y seguimiento de personas que está descrita en el Capítulo 3 y la segunda que trata el tema del reconocimiento de interés en una interacción y la medida de ciertos tipos de respuesta humana que están detallados en los Capítulos 4 y 5.

Las principales aportaciones de esta tesis doctoral son las siguientes:

- El desarrollo de un algoritmo de seguimiento estéreo que utiliza una medida de confianza. La medida de confianza se utiliza para modificar la distribución de probabilidad de los pesos de las partículas en el algoritmo de filtro de partículas. Esta propuesta es rápida, robusta y además permite manejar la incertidumbre asociada a la información de disparidad.

- El desarrollo de un algoritmo difuso de seguimiento estéreo. En esta propuesta no sólo se trata la incertidumbre asociada a la disparidad sino que también se considera la del resto de fuentes de información.

- Un nuevo sistema difuso que permite la detección visual de demandas de interacción. Se calcula un nivel de interés en tiempo real usando un enfoque basado en imágenes y Máquinas de soporte vectorial.

- La propuesta de un sistema difuso jerárquico para medir la respuesta humana usando visión estéreo. El sistema difuso jerárquico es capaz de tratar con la incertidumbre e imprecisión de las medidas en función de la distancia a la que se encuentra la persona.

En el Capítulo 3, se han mostrado dos métodos de detección y seguimiento de personas, uno basado en un enfoque probabilístico (descrito en la sección 3.1) y otro basado en un enfoque "posibilístico" (sección 3.2). Ambos combinan diferentes elementos visuales y utilizan filtros de partículas. Ambos emplean el concepto de proyección de una persona que está formado por dos elipses: una para la cabeza de la persona y otra para su pecho.

En el enfoque probabilístico, las partículas del filtro representan posibles posiciones en 3D del modelo que son evaluadas después de ser proyectadas en la imagen de la cámara. Este método integra información de color, profundidad y de gradiente de manera que es capaz de realizar una seguimiento robusto. Como la información de profundidad no siempre puede ser obtenida debido a las situaciones de oclusión y de falta de textura, el método probabilístico trata este problema por medio de una medida de certeza que indica el grado de confianza en la información de profundidad. La medida de confianza se emplea para modificar la función de distribución probabilística utilizada en el momento de evaluar las partículas. Cuanta más información de profundidad se puede utilizar para una determinada partícula, mayor será la contribución de la información de profundidad de esa partícula y viceversa. En el caso extremo en que no haya disparidad, el seguimiento se basa solamente en la información de color y gradiente. El algoritmo propuesto, no solo determina la posición 3D de la persona, sino también la posición de su cabeza en la imagen de la cámara.

La validez de la propuesta se ha validado usando diversas secuencias de videos con información de color y profundidad. En las secuencias aparecen un número distinto de personas (1 a 4 personas) que interaccionan en una sala. En esas secuencias, las personas ejecutan varios tipos de interacción: desde andar a diferentes distancias, apretar las manos (saludar), cruzar sus caminos, saltar, correr, abrazarse hasta cambiar rápidamente de posición intentando confundir al sistema. Los errores de seguimiento se han calculado para diferentes números de partículas a fin de calcular el adecuado número de partículas que permite un buen equilibrio entre error de seguimiento y tiempo de procesamiento. Los resultados experimentales muestran que el método propuesto es capaz de determinar, en tiempo real, las posiciones 3D (de las personas) y 2D (de la cara en la imagen de la cámara) de una persona que se está moviendo, a pesar de la presencia de otras personas. Además, el método es capaz de resolver adecuadamente oclusiones parciales y/o momentáneas.

El enfoque "posibilístico" intenta resolver algunas situaciones que no han sido tenidas en cuenta en el enfoque probabilístico. En primer lugar, en la propuesta anterior, hay ciertos casos en que el color del background puede ser confundido con el color de la persona que está siendo seguida. En esta segunda propuesta, se tiene en cuenta la información de background y de foreground, lo que permite evitar algunas situaciones confusas. En segundo lugar, la propuesta probabilística no tiene en cuenta algunas situaciones de oclusión que sí se tienen en cuenta en la propuesta ”posibilística”. En tercer lugar, en la propuesta probabilística, existe solo una medida de confianza basada en la información de disparidad. En la segunda propuesta, no solo se tiene en cuenta la información de disparidad para calcular el nivel de confianza, sino también la distancia a la que se encuentra la persona y la posibilidad de estar ocluida o no. Se considera que, si la persona se encuentra parcial o completamente ocluida, la confianza en la información disponible de color y de disparidad disminuye.

Además de estas ventajas, se presenta un método más elaborado para la detección de personas, con el objetivo de reducir al máximo el número de falsos positivos. Asimismo, el error de tracking 2D (cabeza) es menor en el segundo método.

Las ventajas comentadas están principalmente relacionadas con el aumento de fuentes de información con respecto al primer método. Esto es posible gracias a la utilización de la lógica difusa que es capaz de tratar gran cantidad de información de una manera sencilla. Así, el uso de lógica difusa en la evaluación de cada partícula tiene ciertas ventajas con respecto al enfoque probabilístico. En primer lugar, cuando se utiliza un modelo probabilístico para evaluar las partículas estamos asumiendo que las variables siguen una distribución probabilística. Para conseguir eso, la incertidumbre se modela modificando una distribución probabilística a través de unos parámetros. Estas suposiciones no siempre corresponden exactamente a la realidad y no son fáciles de ser modeladas en forma probabilística. Sin embargo, la lógica difusa permite alcanzar el mismo objetivo de una manera más flexible, sin estar sujeta a las restricciones de un modelo probabilístico. En segundo lugar, la lógica difusa permite un incremento gradual de otras fuentes de información de manera sencilla. Al utilizar variables lingüísticas y reglas para expresar relaciones entre las diferentes fuentes de información, el sistema se convierte en un sistema más comprensible y similar a la interpretación humana del conocimiento.

Por otro lado, al utilizar más fuentes de información aparece el inconveniente de necesitar más tiempo de ejecución para obtener el resultado final. Sin embargo, y aunque el tiempo de ejecución del segundo método es superior al del primero, el segundo está preparado para tratar situaciones más complejas.

El segundo método ha sido comparado experimentalmente con otros métodos de seguimiento bien conocidos en este campo. Los resultados muestran que nuestro sistema ha logrado mantener el seguimiento de las personas, en la imagen de referencia de la cámara, en gran parte de las situaciones donde los otros métodos fallan. Se ha probado en situaciones complejas que simulan la vida real, en las que personas se encontraban interaccionando de manera libre y ocluyéndose, a veces, unas a otras. Se ha comprobado experimentalmente que el segundo método es suficientemente rápido en la detección y seguimiento de personas y, consecuentemente, adecuado para aplicaciones en tiempo real.

En el Capítulo 4, se presenta un sistema capaz de estimar el interés de las personas que se encuentran en el entorno del robot y también de detectar determinados movimientos de brazos que indican una demanda de atención o movimientos de la cabeza que indican concordancia o discordancia. El método utiliza visión estereoscópica y estimación de la orientación de la cabeza a través de SVM y lógica difusa. Mientras se sigue a una persona, el sistema difuso calcula el grado de interés que tiene la persona en interaccionar con el robot. Este valor de interés se basa en la posición de la persona con respecto al robot así como en la estimación de la atención que esa persona está prestando al robot. Para calcular la atención, la orientación de la cabeza es estimada en tiempo real. Este análisis se hace utilizando un enfoque basado en visión y SVM. Gracias a la SVM, la detección de la orientación de la cabeza se realiza con un gran porcentaje de éxito, e independientemente de las características morfológicas. El sistema difuso empleado también ha sido capaz de detectar de manera precisa cuando la persona se encontraba reclamando la atención del robot con los brazos, así como ha logrado obtener buenos resultados en la detección de movimientos específicos de la cara como los de concordancia y discordancia.

En el Capítulo 5, se calcula la respuesta humana en un modo diferente al Capítulo 4. En este caso no se exige a las personas que cumplan todas las condiciones anteriores al mismo tiempo, es decir, que se encuentran más o menos cercanas al robot, más o menos centradas con respecto a él y mirándolo siempre directamente. La idea es mejorar el análisis del comportamiento humano, por medio de una medida de respuesta humana basada en diferentes elementos visuales, que tienen mayor o menor importancia de acuerdo con la distancia a la cual se encuentra la persona respecto al robot. Estos elementos visuales son la atención que cada persona presta al robot (orientación de la cabeza con respecto al robot), la oclusión (considerando que una persona intenta evitar los obstáculos entre el robot y ella misma durante una interacción), ciertos tipos de movimientos con los brazos (ya que el robot puede pedir a las personas que interaccionen agitando los brazos) y la detección de la sonrisa (que debe transmitir una buena idea del grado de satisfacción de las personas con respecto a ciertos mensajes y acciones propuestos por el robot). La lógica difusa será, una vez más, la herramienta utilizada para fusionar todas estas fuentes de información, ya que permite un tratamiento fácil de la incertidumbre e imprecisión que puede existir en la información obtenida a partir de los sensores. Se propone un sistema difuso jerárquico de manera que un sistema difuso de alto nivel trata las salidas de otros dos sistemas difusos de más bajo nivel, teniendo en cuenta la distancia a la que se encuentra la persona respecto del robot. Cada uno de los sistemas difusos de bajo nivel está especializado en la situación de distancia cercana o lejana de la persona con respecto al robot. El sistema difuso de alto nivel utiliza la salida de los sistemas difusos de bajo nivel en función de la distancia de la persona de manera que la información visual se utiliza de la forma más conveniente para medir la respuesta humana.

A partir de la salida del sistema difuso se proponen diferentes medidas de la respuesta humana. Estas medidas son la respuesta humana instantánea HR(P), la respuesta humana global GHR(P) y la media de HR(P). Al tomar en cuenta valores instantáneos, globales y medios podemos analizar que aspectos de la actividad propuesta por el robot se pueden mejorar. Por ejemplo, si la retroalimentación obtenida por parte de los usuarios no es satisfactorio el robot podría cambiar su forma de comportarse durante la actividad de interacción social. El sistema se ha probado en situaciones que simulan la vida real, en las cuales se ha pedido a las personas que participaban que se moviesen y actuaran de una manera natural. A pesar de la gran cantidad de información analizada hemos observado que el algoritmo tiene un buen rendimiento para ser utilizado en situaciones de la vida real, permitiendo una interacción natural entre todas las personas que participan en los experimentos. Además, el sistema difuso jerárquico calcula los valores esperados de respuesta humana en casi la totalidad de los frames analizados.

Como conclusión, y como se indicó en el Resumen inicial, pensamos que los trabajos presentados en esta tesis, que han sido implementados y probados en condiciones de la vida real, contribuyen en diferentes aspectos al desarrollo de este área de investigación.

En los diferentes sistemas difusos descritos en este trabajo las reglas y las variables lingüísticas se han definido de manera experimental. Como trabajo futuro, se está estudiando la posibilidad de construir un sistema capaz de aprender y ajustar esos parámetros de forma automática. Otra cuestión a tener en cuenta para trabajos futuros es el aprovechamiento de la retroalimentación de los usuarios con respecto a las actividades propuestas por el robot para mejorar dichas actividades y/o la interacción que se produce entre el robot y los usuarios. Por último, hay que tener en cuenta que la modularidad del sistema permite la incorporación de otras fuentes de información. En este sentido, se está considerando la incorporación de sensores de sonido y de técnicas de reconocimiento de voz como posibles mejoras del sistema en trabajos futuros.