TECHNICAL UNIVERSITY OF CRETE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

DEVELOPMENT OF A SENSOR FOR DETECTING THE POSITION OF OBJECTS IN SPACE IN VIRTUAL REALITY APPLICATIONS

Konstantinos-Georgios Tryfonas
3/9/2016

TECHNICAL UNIVERSITY OF CRETE
SCHOOL OF
ELECTRICAL AND COMPUTER ENGINEERING

DEVELOPMENT OF A SENSOR FOR DETECTING THE POSITION OF OBJECTS IN SPACE IN VIRTUAL REALITY APPLICATIONS

Examination Committee:

Eftychios Koutroulis – Associate Professor (supervisor)

Apostolos Dollas – Professor

Katerina Mania – Associate Professor

Konstantinos-Georgios Tryfonas

Chania 2016

Acknowledgements

I would like to thank Associate Professor Eftychios Koutroulis, for giving me the opportunity to work on this thesis and providing fruitful suggestions. Also, I would like to thank the Examination Committee for their participation. Last but not least, I would like to sincerely thank my family for their support during my studies at the Technical University of Crete.

Abstract

Virtual Reality (VR) and Augmented Reality (AR) systems have entered our lives. During the past decade they were used only for research purposes, or in simulation environments, but nowadays enterprises invest in mass production of VRs for home entertainment as well. In Virtual Reality, a computer reproduces an alternative environment and places the user in this simulated immersive world, giving her/him the experience of a different reality. The feeling of physical presence in the simulated world is created artificially through vision, touching and hearing. In Virtual Reality, the position and orientation of tracked objects are measured by a complementary sensing system that is solely responsible for positional tracking. The necessity of positional tracking is a major drawback for VR systems, because it requires the employment of dedicated sensing infrastructure, rendering these systems expensive.

In this thesis, a low-cost, portable sensing system for positional tracking was developed, by using an accessory of the commercially-available console Wii, called Wiimote (or Wii-remote), that is manufactured by Nintendo. This device contains a 3-axis accelerometer and a high-resolution high-speed infrared (IR) camera and utilizes a Bluetooth interface for connecting with other devices. Two Wiimotes and IR LEDs were used to provide the necessary means for calculating the position of the user. The two Wiimote cameras are installed in static, known locations and an IR-LED bulb is mounted atop of the user. A software application was also developed, which is capable of providing real-time coordinates of the user to the actual VR system with adequate accuracy. The position tracking system that was developed is affordable for home users, since it is easy to deploy and inexpensive.

Table of Contents

1. Introduction

Following the achievements of computational geometry and computer vision in the past years, the need to provide the user of a virtual reality system with absolute realism has become increasingly important. In Virtual Reality (VR), a computer reproduces an alternative environment. The feeling of physical presence in the simulated world is created by artificial senses which may incorporate sight, touch, hearing and smell. Therefore, for giving the experience of existence in the simulated environment, an indoor positioning system is deployed for providing real-time coordinates of the user to the actual VR system. Moving around the scenery, provided by Head Mounted Displays (HMDs), is a key feature for immersive virtual reality.

An Indoor Positioning System (IPS) is a system responsible for providing the location of people (or objects) in the form of coordinates in enclosed environments, such as the inside of buildings, where Global Positioning Systems (GPSs) cannot operate due to weak signal and interference. IPS systems mainly utilize radio waves, acoustic signals or magnetic fields.

In the past few years several such systems were created and there are a lot of commercial systems on the market. However, it has by now become evident that we cannot rely on a single technology for designing that kind of systems, in contrast to outdoors positioning which is solely provided by satellite-based localization. IPS systems differ on many aspects. The reason behind this is that user requirements play a much greater role in these systems.

1.1 Applicability and User requirements

Indoor positioning has grown to be one of the most rapidly evolved technological fields over the past decade. Day by day, businesses endow massive amounts of money for positioning systems. Present day lifestyle involves location-based applications to aid us in everyday life accomplishments. The development of the current systems is closely tied to the different requirements determined by each application. Consequently, the need for dedicated local infrastructure for addressing the requirements set by the application is indubitable. Therefore, it is necessary to list the performance parameters of these kinds of systems. Below we present a quick list of applications that require tracking:

- *Location-based services in indoor environments* (e.g. warehouses, train stations etc.)
- *Private homes* (e.g. ambient assistant living systems for elderly people, medical monitoring etc.)
- *Context detection and situational awareness* (e.g. car collision, assisted parking etc.)
- *Medical care* (e.g. robotic assistance in surgeries, patient tracking in hospitals etc.)
- *Triggered context aware information services* (e.g. museums, parks etc.)
- *Motion capturing* (e.g. movies, video games etc.)
- *Computer vision* (e.g. virtual reality, augmented reality, robotics etc.)

Each field listed has different prerequisites for position detection. It is obvious that location-based services in warehouses should not share the coverage and accuracy of positioning in robotic assistance surgeries. As such, an essential factor for creating any IPS system requires a meticulous examination of the user requirements. A list of the most important user requirements contains:

- ✓ *Accuracy* (e.g. mm, cm, meters etc.)
- ✓ *Coverage area* (e.g. single room, building etc.)
- ✓ *Manufacturing and maintenance costs*
- ✓ *Update rate* (on-event, on request or periodically e.g. 100 Hz, or once a week etc.)
- ✓ *Interface* (e.g. text-based or through GUIs)
- ✓ *Robustness* (in terms of e.g. physical damage, theft, jamming, unauthorized access etc.)
- ✓ *Scalability*
- ✓ *Latency*
- ✓ *Intrusiveness* (disturbing, imperceptible)
- ✓ *Privacy* (active or passive devices, mobile or server-based computation)

The diversity of the technologies used in indoor position detection systems arises from the importance of each aspect, given the domain and user requirements. For each specific application, the performance parameters should be coordinated with the user prerequisites, in order to decide upon a suitable positioning technology.

1.2 Motivation and Background

Entertainment applications based on Virtual Reality (VR) and Augmented Reality (AR) require position detection systems with high update rates, high precision and lightness. If we add up the requirements specifically for home users (e.g. robustness, ease of deployment etc.), the challenge becomes harder. Meeting all of these requirements often leads to high costs, rendering virtual reality for home users inaccessible.

Considering the above, with emphasis on a low-cost, portable solution able to cope with human motion, operate in combination with any HMD system and provide the user with a relative good precision and coverage of a room-sized environment for home or laboratory use, an InfraRed (IR) camera scenario was adopted in this thesis. IR positioning is another possibility for indoor localization and this technique has been extensively used in smart items or mobile robot localization applications.

IR positioning appertains to the class of “optical indoor positioning” techniques, where a camera is the only or the main sensor. The usage of optical methods in positioning has grown to be one of the most common solutions. The reason behind their success mainly comes from the improvement of detectors (e.g. CCD sensors). Moreover, there has been a great progress in the development of algorithms in image processing, facilitating higher data transmission rates.

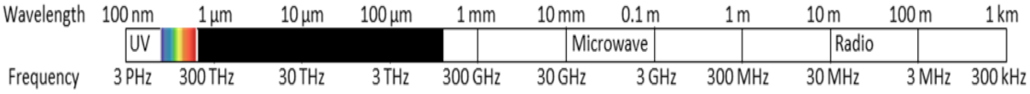


Fig. 1.1 Infrared radiation: wavelength 700 nm-1 mm, frequency: 430 THz-300 GHz [1].

As shown in Fig. 1.1, IR wavelengths are longer than that of visible light, but shorter than that of Terahertz radiation. Therefore, IR radiation is undetectable to the human eye. IR technology is commonly used as an alternative to visible light in indoor positioning, due to its non-obtrusive nature. Furthermore, few things in an ordinary house radiate strong infrared light, rendering it perfect for use in room-level environments.

This solution has been used extensively by many high-precision systems [2] that offer professional standards. IR cameras are regular cameras with a band-pass filter, which allows only light in the infrared part of the spectrum to be analyzed. Even so, to monitor human motion, high update rates are required. However, IR cameras with high update rates and a high resolution are quite expensive. Actually, their prices are ranging from three hundred to several hundreds of dollars, depending on their operational characteristics.

As mentioned, the goal of this thesis was to create an affordable system without the need of dedicated infrastructure and complicated installation. After a thorough research, an alternative solution was considered. Nintendo Corporation, on the 19th November of 2006, released a home gaming console named Nintendo Wii. The Wii was registered as the best-selling home console of its generation. The reason behind its success was based on its innovative controller. This handheld device, namely Wiimote or Wii-remote, was able to track hand movements. At that time it was a breakthrough in the game industry due to its low cost (its suggested price was 30 €). Wiimote is capable of detecting and measuring fast movements due to its high update rate and high resolution of its embedded IR camera, rendering it ideal for human movement recognition. Real-time computer vision based on a regular webcam with typical resolution of 640 x 480 at 30 Hz refresh rate can be easily outperformed by the characteristics provided by the IR camera of a Wiimote. It should also be noted that webcams require a high CPU computing power when they perform real-time tracking compared to Wiimotes. Furthermore, the popularity and the large-scale manufacturing of the Wiimote controller brought down its retail price.

The majority of optical tracking systems are based on the analysis of 2D images gathered from projections. Another way to estimate the position and orientation of an object is through sweep-beam angle determination. Optical systems can be classified into “outside-in” and “inside-out” based on the way the camera is placed in relation to the given target. In detail, when the camera is placed on the object being tracked it is called “inside-out” because the camera is “looking out into” the world. Contrary, when the camera is placed externally and it is watching the target from a distance, it is called “outside-in”.

In our case, by combining two Wiimotes and using them in the opposite way, hence like IR cameras in an “outside-in” manner, we were able to track a homemade IR-bulb that contained an array of IR-LEDs and measure 3D positions in space as shown in Fig. 1.2. As mentioned before, our system requires two Wiimotes installed in parallel to each other. Wiimotes must have a clear optical sight of the IR-bulb that is mounted on top of the user wearing the VR glasses or the HMD. The IR stimulus is reported back from the two different Wiimote camera locations, in pixel form. Then, by acquiring the gathered data through a Bluetooth communication link between the Wiimotes and the PC responsible for VR processing, we solve the triangulation problem and report the coordinates of the user in units, assuming as the origin of coordinates the left Wiimote camera CCD position in space. The scenario will be explained in detail later on.

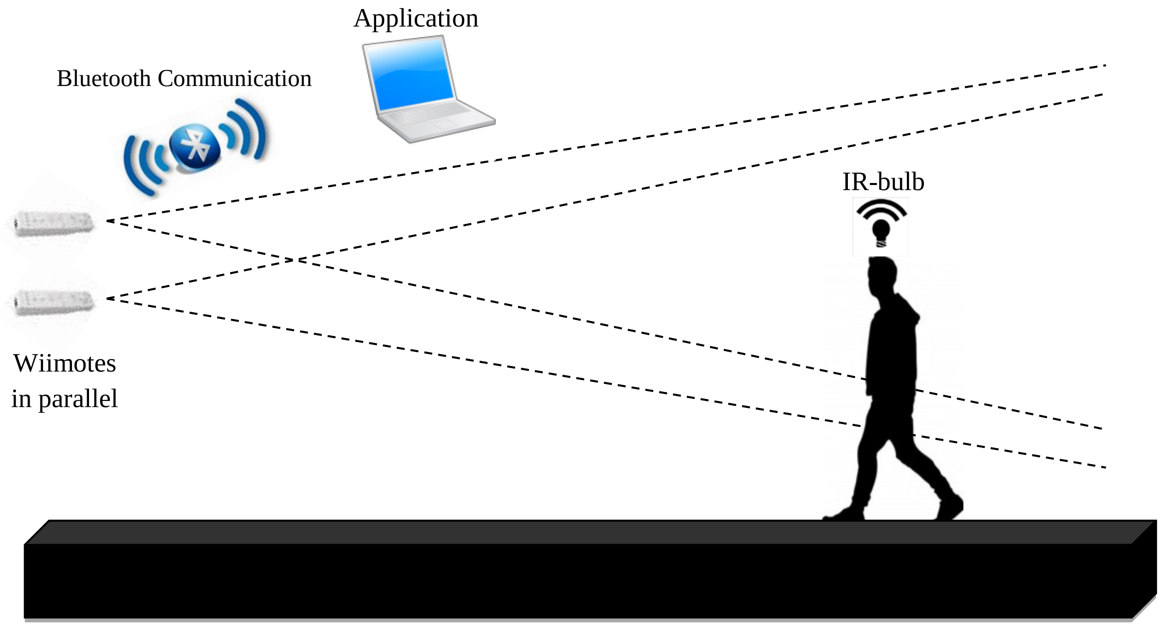


Fig. 1.2 Abstract graphical view of the proposed system: the user walking through space with the mouted IR-bulb on top of his head;the pixel coordinates are transferred through a Bluetooth link to a central PC for position calculation.

1.3 Thesis Structure

The introduction is followed by an overview of the different technologies used in IPS systems in general. In Chapter 2 the reader will be informed briefly for the majority of technologies already utilized. First, we make a reference to the positional methods and basic techniques in the field and later we present the major technologies commonly used, according to the category they fit into. The proposed system is described in Chapter 3. We start by listing the key features of Wiimotes and the other hardware that was used. After that, we define the technical terms frequently used in stereo vision. The end of Chapter 3 is devoted to a more detailed presentation of the scenario adopted. Last but not least we present the software tools and the actual implementation of the application created for providing coordinates of the user to the actual VR system. Then, experimental results are presented in Chapter 4, where the accuracy and the coverage characteristics of the implemented system were investigated. Chapter 5 summarizes the findings and provides suggestions for improving the proposed system performance according to the general conclusions drawn from the presented method.

2. Overview of Technologies

The performance of outdoor positioning systems has been improved at an exceptional level, due to the development of worldwide satellite positioning. However, inside positioning cannot be based on the same technologies utilized by outdoors positioning systems. Therefore it is essential to divide positioning into outdoors and indoors.

Indoor positioning has progressively turned into a center of examination since advances for positioning in outdoor environments (such as the Global Navigation Satellite Systems, GNSS) perform ineffectively in the inside of structures, because of deficient view of the open sky. Thus, the ability to locate objects and people inside buildings remains a considerable challenge because of limitations due to restrictions in positioning determination, resolution, system dynamic responses and complicated arrangement of the internal space (e.g. due to furniture, inside walls etc.) which impede the task of position measurement. An impeccable positioning technique for indoor applications has not yet been developed. Different solutions currently exist for indoor positioning and navigation, such as Inertial Navigation Systems (INS), ultrasonic localization, laser range finders, as well as radio-signal-based techniques, such as Received Signal Strength Indication (RSSI) method, each having their own strengths and weaknesses.

2.1 The Basic Position Measuring Principles

The most common techniques used in the field are given below. These measuring principles are the bare bone of almost any positioning method.

Time of Arrival (ToA) / Time of Flight (ToF)

Time of Arrival (ToA), sometimes called Time of Flight (ToF), is the travel time of a radio signal from a single transmitter to a remote single receiver (Fig. 2.1). The distance between the two can be derived by the multiplication of the signal travel time by the wave speed. Since the wave speed relies upon the properties of the propagation medium, information of the intermediate material (e.g. air, concrete etc.) is required. Synchronization of transmitter and collector clocks is important, as even one nanosecond blunder in synchronization translates into a distance error of 30 cm if radio frequency signals are utilized. ToA is especially hard to apply in indoor applications, where multi-path conditions are regular. The utilization of a wider frequency band is an approach to address this issue.

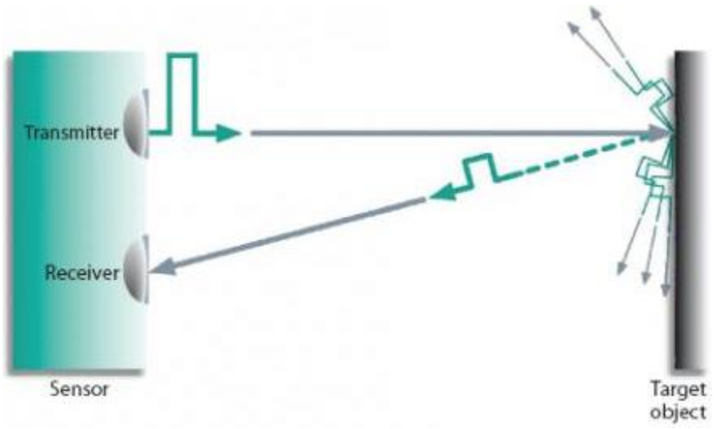


Fig. 2.1 Basic concept of ToF technology distance determination [3].

Time Difference of Arrival (TDoA)

TDoA, as the name implies, follows the same tactic as ToA with the variance that in TDoA the receiver does not need to know the absolute time at which a pulse was transmitted, just the time difference of arrival from the synchronized transmitters. With that in mind, constant time offset caused by a clock bias is subtracted from ToA measurements. With two emitters at known locations, a receiver can be located onto a hyperboloid. A receiver's location can be determined in 3D from four emitters by intersection of three hyperboloids. With this methodology, very precise synchronization of all emitters is a precondition. For GNSS positioning TDoA is a useful approach, because the drift of a low-cost receiver's clock can be eliminated while the satellites are precisely synchronized by 'GNSS time'. Alternatively, a portable emitter can be located from numerous receivers. In this arrangement, the static nodes-receivers are attempting to decide the location of the portable station.

Round Trip Time (RTT) / Round-trip Time-of-Flight (RToF) / Two Way Ranging (TWR)

RTT, otherwise Two-Way Ranging (TWR), is the time required for a signal pulse to make a trip from a specific source to a specific destination and back again. RTT eliminates the need of time synchronization between the transmitter and the receiver, permitting its use in uncoordinated mesh-like networks. As a disadvantage, critical latencies may occur because the range estimations to multiple devices should be completed consecutively for applications where devices move rapidly. However, this method has the benefit of low complexity and cost.

Phase of Arrival (PoA) / Phase Difference (PD)

PoA utilizes the received carrier phase to decide the separation between two devices. Keeping in mind the end goal to mitigate phase wrapping, the received signal phase is assessed on multiple frequencies. The distance is then determined by the rate of phase change.

Near-Field Electromagnetic Ranging (NFER)

NFER refers to any radio technology utilizing near-field properties of radio waves. It employs transmitter tags and one or more receiving units. The phase of an electro-magnetic field changes according to the distance around an antenna. Depending on the choice of frequency, NFER has potential estimations in the precision scope of 30 cm to 1 m and operating distances up to 300 m.

Angle of Arrival (AoA)/Angulation / Triangulation / Direction-based Positioning

Angle of arrival (AoA) is a method for determining the direction of propagation of a radio-frequency wave incident, on a direction sensitive antenna array (Fig. 2.2). AoA determines the direction by measuring the Time Difference of Arrival (TDOA) at individual elements of the array. By measuring these delays, the AoA can be calculated.

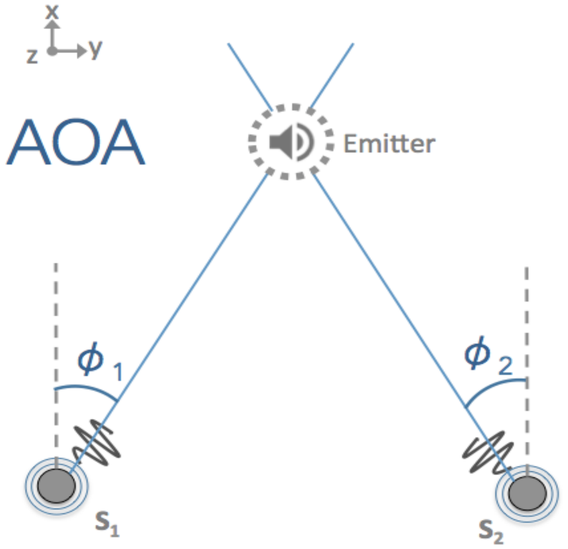


Fig. 2.2 An Angle of Arrival (φ) is calculated, representing the direction from which the received signal was initially emitted [4].

Doppler Ranging

The Doppler ranging method uses the Doppler Effect to produce velocity data about objects at a distance. It does this by bouncing a microwave signal off a desired target and analyzing how the object's motion has altered the frequency of the received signal. Given a known starting position and different Doppler frequency observations, the displacements of a mobile device can be resolved through the measured Doppler frequency shifts (Fig. 2.3).

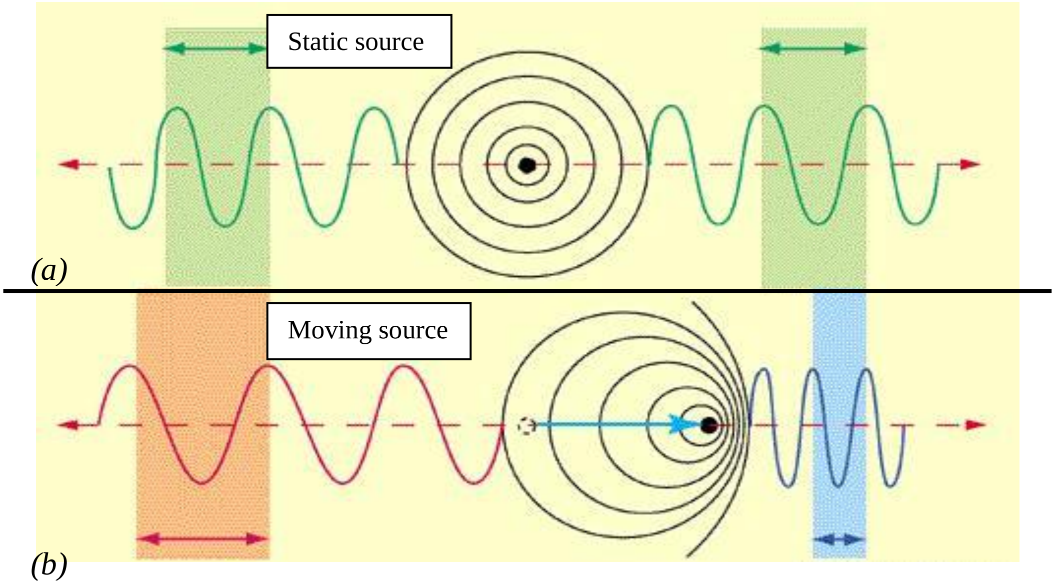


Fig. 2.3 Doppler Effect: (a) A static wave source relative to its wave crests. (b) The same wave source moving right. Assuming we observe a wave source moving towards a direction, the apparent wavelength of the source changes relative to its motion [5].

Received Signal Strength Indicator (RSSI)

In this method, as the name indicates, signal attenuation is used to provide distance estimation based on how far a beacon is from the signal source. By sampling the received signal strength values of source over a time period and calculating its average we are able to produce an RSSI weight mapping of a certain area-volume. RSSI values are measured in decibels, because they are specified as power measurements. Given a signal source, RSSI range circles are formed in relation to the source based on the integrated signal attenuation model.

2.2 Positioning Methods

Following the basic measuring principles used in positioning, we introduce the most common positioning methods used in the field. The determination of position of an object is given through measurements of proximity, distance and angular information.

Cell of Origin (CoO) / Proximity Detection / Connectivity-Based Positioning

CoO method determines the position of a target by its existence in a particular area. The location of the base station is ascertained and the location of the caller is considered to be where the strongest signal is received. CoO is not a very precise locator and its accuracy depends on the number of base stations in the search area and the signal range. CoO is a simple positioning method used for applications with low requirements in terms of accuracy. Examples are sensors detecting physical contact and mobile wireless positioning systems.

Centroid Determination

Given a set of multiple beacon positions (e.g. Wi-Fi transmitters), the centroid beacon is to be located. The most common method of centroid determination is through RSSI values. The position of the centroid beacon can be determined by the weights in form of RSSI values measured by the centroid beacon. Knowledge of the positions of each beacon within the detection area is a requirement of this technique.

Lateration / Trilateration / Multilateration

Each of these terms contains the process of determining absolute or relative locations of points by measurement of distances to nearby nodes (Fig. 2.4). The most common techniques used for distance measuring are the RSSI, ToA, TDoA and RTT.

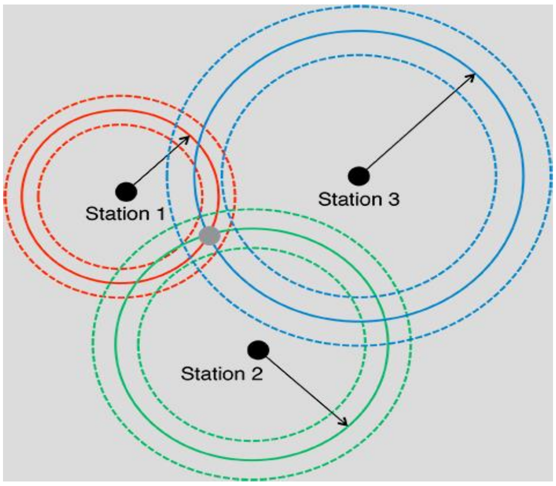


Fig. 2.4 The multilateration method: the mobile node (grey dot) position is calculated from the intersection of the known RSSI range circles of the static nodes [6].

Polar Point Method / Range-Bearing Positioning

In this method, there is only one station capable of measuring distance and angle for determining the coordinates of another station given the fact that orientation of the main/first station is already known. The polar point method is commonly used in geodesy because one station can determine the position of multiple targets, without the need of complicate setup of multiple stations (Fig. 2.5).

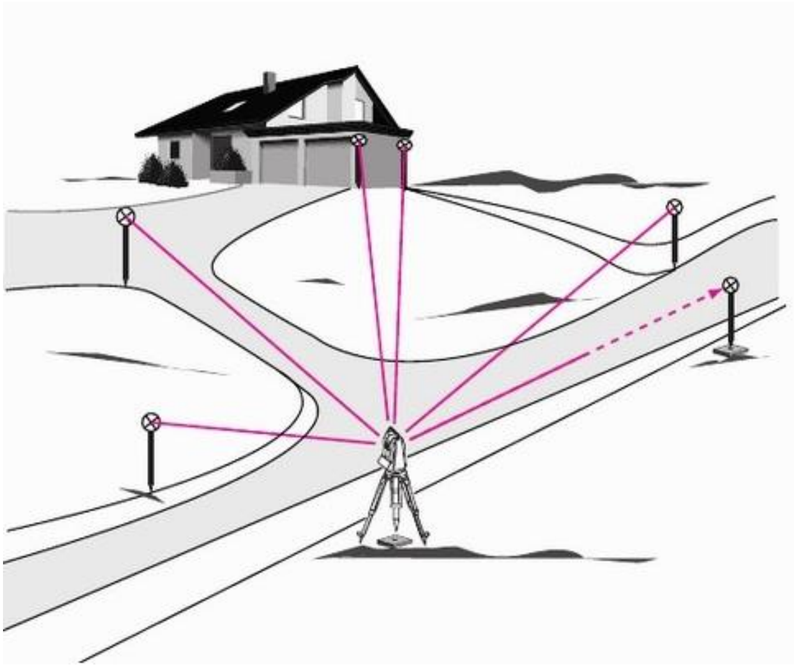


Fig. 2.5 The polar point surveying method: the position and height of a point on the object are determined by measuring angles and distances [7].

Fingerprinting (FP) / Scene Analysis / Pattern Matching

Fingerprinting is a procedure that maps data (such as RSSI values, images or audio signals) that are uniquely identified, just as human fingerprints uniquely identify people for practical purposes. The fingerprinting algorithm has two phases; the off-line calibration stage and the on-line measurement stage. In the first stage, a database with all the unique fingerprints is created, mapping the scene. In the second stage, the fingerprint measured by the online phase is compared with the database created in the first stage and the best fit is computed. Assuming we scatter multiple Wi-Fi or Bluetooth beacons across a building or room, we create a database with the measured RSSI values taken in each room of the building (Fig. 2.6). In the online phase, the current RSSI value is compared with the values gathered and the best matching/fit reveals the coordinates.

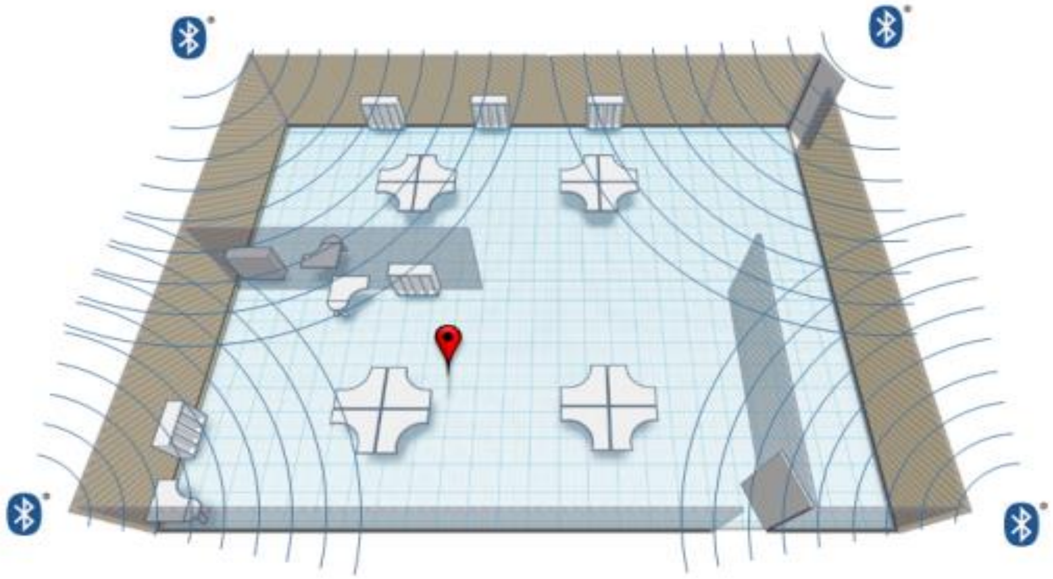


Fig. 2.6 Bluetooth fingerprinting based on RSSI value: the location of the target is calculated by the signal strength measured by each beacon [8].

Dead reckoning (DR)

The term "Dead Reckoning" (DR) is the process of estimating the value of any variable quantity by using an earlier value and adding whatever changes have occurred in the meantime (Fig. 2.7). A system of inertial navigation uses DR to estimate current position by measuring the speed over an elapsed time in relation to its starting position. The sensor types used in DR are accelometers and gyroscopes. Dead reckoning is subject to significant errors as both speed and direction must be accurately known at all instants for position to be determined accurately. Last but not least, inaccuracies produced in the process are cumulative, adding up to the error deviation as time goes by.

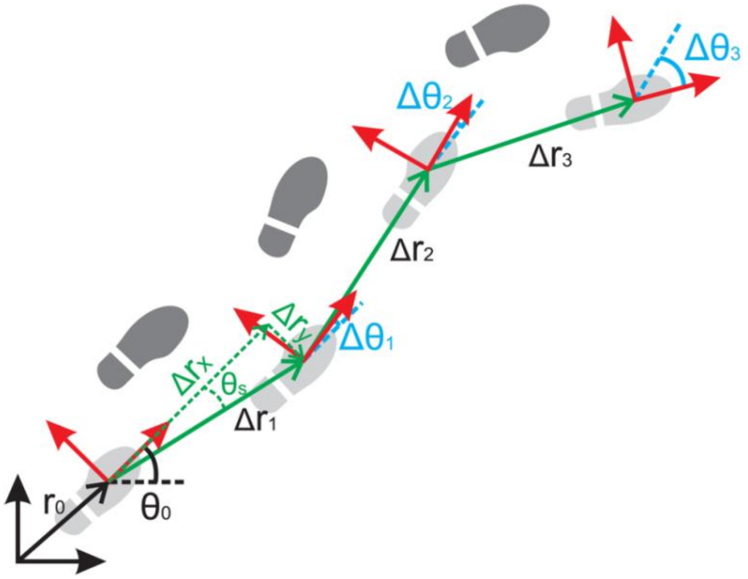


Fig. 2.7 Human motion tracking based on internal motion: the current position is calculated by adding whatever changes have occurred during the course of time [9].

Kalman Filter (KF) and Extended Kalman Filter (EKF)

Kalman filtering, also known as linear quadratic estimation (LQE), is an algorithm that combines Bayesian inference and joint probability distribution over a series of measurements observed over time containing statistical noise or other inaccuracies for producing a more precise outcome. The estimates produced tend to be more precise than those based on a single measurement alone. The classic KF is regularly used for camera based tracking and pedestrian navigation based on inertial measurements, for predicting positions in-between data samples given the past history of the model used. The Extended Kalman Filter (EKF) is used for nonlinear observations that have a high risk of divergence when a linearization process is applied.

Map Matching (MM)

Map matching algorithms “match” the current path of the user with pre-recorded spatial maps. By identifying the logical link between the current coordinates of the user with data taken from a form of Geographic Information System (GIS), the positional accuracy is improved without the need of additional hardware (Fig. 2.8). MM techniques include topological analyses, pattern recognition, or advanced techniques, such as hierarchical fuzzy inference algorithms.

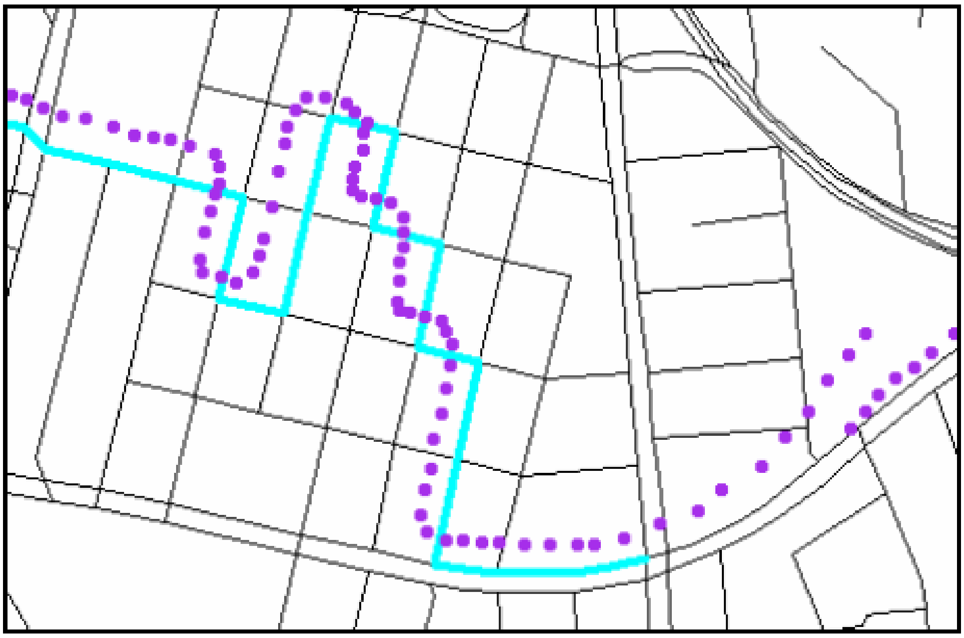


Fig. 2.8 *Example of mismatch detection and recovery on roads utilizing Map Matching. The dotted purple line represents the path measured by the positioning system. The light blue line is the actual path travelled by the user after the utilization of a Map Matching algorithm [10].*

2.3 Technologies used

This chapter describes the most common approaches found in the field of indoors positioning. These technologies illustrate the main types and characteristics of IPS systems. More details about these technologies are provided in [1].

2.3.1 Cameras

Optical indoor positioning utilizes cameras as the main type of sensor. A wide variety of applications with different requirements of accuracy can be produced using cameras. The main reason behind this is the advancement in the technology of detectors (e.g. CCDs) and the decrease in size of actuators (e.g. lasers). However, camera's fundamental application zone is in the sub-mm domain. This means that cameras are typically used in applications requiring millimetric or sub-millimetric accuracy. With the advancement of image processing algorithms and faster data transmission rates, optical methods have become the dominant technique of positioning.

There are two main types of optical indoor positioning systems involving cameras. In the first method, a mobile camera mounted on the user is to be located by reference images captured by it. Comparison between the current image and the pre-gathered reference images reveals the camera location. In the second method, static cameras capture the moving target and turn the 2D observations of the CCD into 3D coordinates. Rotation measurements are based on the Angle of Arrival (AoA) technique. When more than one camera is used, depth information can be calculated by stereo vision algorithms. However, depth acquired from monocular images can be obtained by sequential observations of the scene, using a synthetic stereo vision approach.

The transformation from the CCD image into the actual object location cannot be resolved without additional information regarding the distance. Moreover, the baseline between the cameras must be known beforehand. Laser scanners and range imaging sensors can be used in tandem with cameras for obtaining more accurate distance measurements.

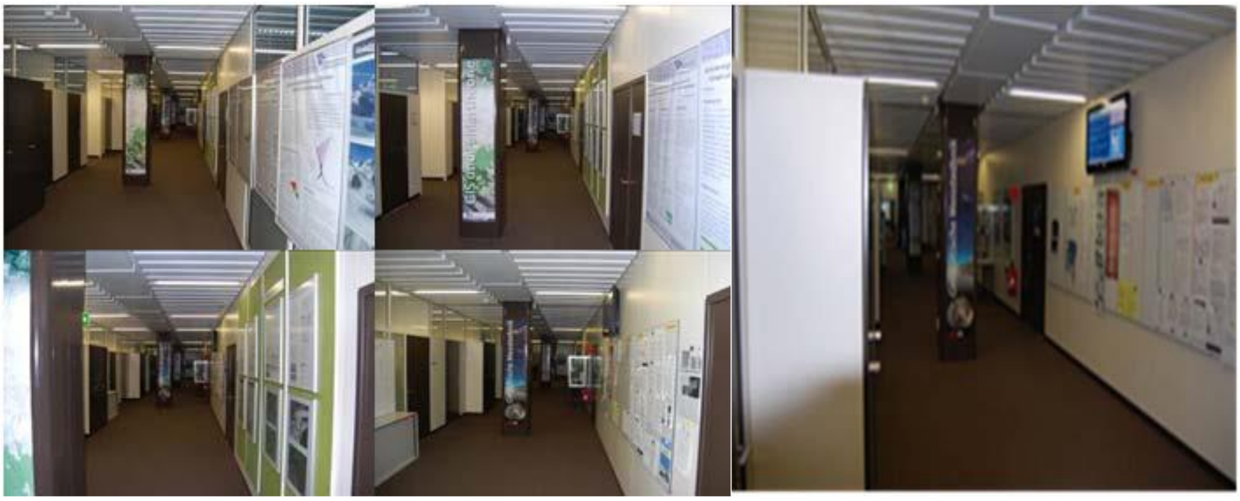
A rough categorization of optical positioning systems can be performed based on how the reference information is obtained, as described next.

Referencing from 3D Building Models

In this category, camera images containing specific objects are compared to interior design databases. From this comparison, location and angle can be detected based on the geometric properties of fixed objects like windows and doors. The main feature of these kinds of methods is that there is no necessity for local infrastructure, because real sensor beacons are substituted by digitally created referencing points through images. In this way, the extra cost imposed by scaling coverage, is eliminated as well.

Reference from Images

This class of positioning methods, also mentioned in the literature as view-based approaches, consists of two stages. In the first stage, sequences of images are gathered along multiple paths inside a building. In the second stage, the current view of a mobile camera is compared to the previously captured view sequences (Fig. 2.9). As a drawback, an external reference source must be utilized sporadically to regulate accumulated deviations.



(a) (b)

Fig. 2.9 (a) Example of a view sequence. (b) Current view to be compared with the view sequence [1].

Reference from Deployed Coded Targets

Another commonly used method of optical positioning is the deployment of unique markers (Fig. 2.10). These markers are scattered throughout the coverage area and uniquely stamp the desired regions. Coded markers can be placed in different locations (e.g. floor, walls) depending on the way they are being recognized by the sensor. Compared to systems relying entirely on natural features of CCD images, coded targets are more reliable under conditions of varying illumination. As a result, the level of accuracy from the reference points is higher, making it suitable for positioning systems demanding higher accuracy requirements. Last but not least, they simplify the process of image detection thanks to the use of well-defined geometrical shapes and coloring.

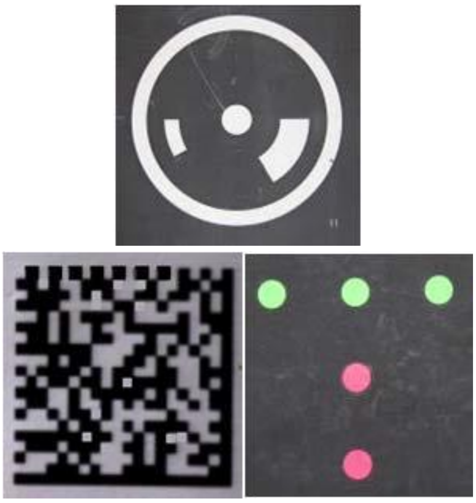


Fig. 2.10 Examples showing different models of coded targets-markers [1].

Reference from projected targets

If mounting of reference markers is not a possible or desirable solution, one can turn to a different approach by projecting reference points or patterns in the desired area from complementary devices such as LEDs or projectors (Fig. 2.11). A common way is to project infrared light in order to attain unobtrusiveness to the user, because the infrared light cannot be perceived by the human eyes. Location of the user is calculated by differentiations in color or shape of the pattern projected. This method eliminates the cost of physical reference targets.

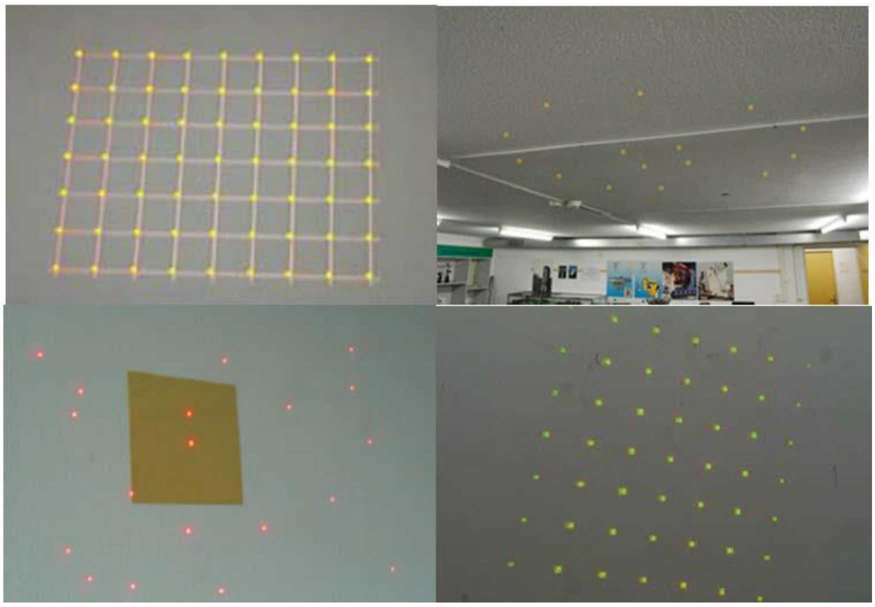


Fig. 2.11 Examples of different types of projected reference patterns [1].

Systems without Reference

In this category fall all systems deploying static cameras with high frame rates for position detection. Real-time tracking is accomplished by observing the differences in images of subsequent frames. This method requires the placement of orientated static cameras on fixed, known, locations.

2.3.2 Infrared

Infrared radiation has been used extensively in positioning systems due to fact that it is invisible to the human eye compared to visible light. There are three major categories that utilize IR radiation to perform position detection: active beacons, IR imaging with natural radiation and IR imaging with artificial light sources.

Active Beacons

The architecture of systems utilizing active beacons consists of placing infrared receivers throughout the coverage area on fixed, known locations. A mobile unit or "active beacon" mounted with an IR emitter is then located based on which receiver picks up its signal. Several IR emitters or beacons with different IR frequencies can be located simultaneously. Angular information can be obtained by installing optical polarizing filters on the IR receivers. Active Beacon positioning has a low accuracy level and is used for room-to-room localization.

Position detection with Natural Infrared Radiation

Natural Infrared Imaging, also known as infrared thermography, measures the surface temperature of an object through its heat emission. Positioning systems with IR sensors operating between 8 μm to 15 μm - long infrared spectrum - are also referred as passive infrared systems. Thermal infrared radiation can be used to track people or objects through their temperature without the need of external components (Fig. 2.12). However, strong infrared radiations from sources like the sun or bright ceiling lights can deteriorate the detection accuracy. Thermal cameras, pyro-electric infrared sensors and thermocouples are the most common detectors for performing passive localization.

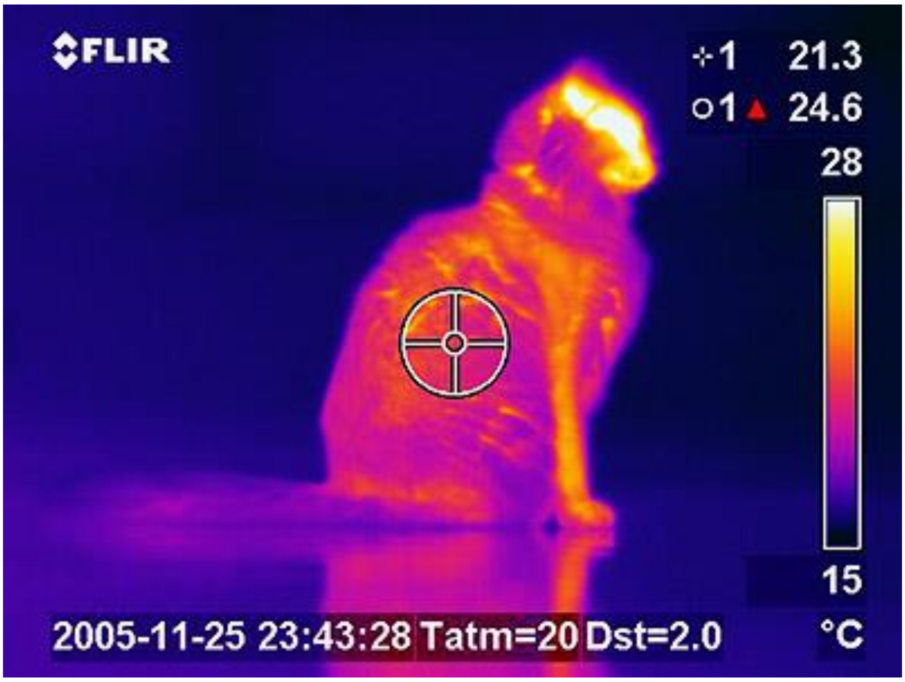


Fig. 2.12 IR thermography of a cat [11].

Position detection with Artificial Infrared Light

Optical IR positioning with artificial IR light imaging involves the use of active IR light sources in tandem with IR-sensitive cameras. This method is commonly used as an alternative to visible light optical positioning systems. Implementations using IR cameras are either based on active infrared LEDs or on retro-reflective targets. In the first approach, the target is mounted with one or several active infrared LEDs and its position is detected by a CCD IR camera. In the second approach, an external IR source illuminates the target and then the reflected light is gathered by the CCD.

Microsoft's well-known "Kinect" device utilizes IR technology (Fig. 2.13). Kinect employs an IR projector capable of covering a room-size space with a predetermined pattern of infrared dots, an IR sensor, which is placed at a given distance relative to the IR projector, capturing the IR dots and a RGB camera. The monochrome IR sensor calculates the distance of the objects captured by its CCD, by subtracting the observed object distance relative to the expected, unobstructed pattern view. People can be tracked simultaneously up to a distance of 3.5 m at a frame rate of 30 Hz. An accuracy of 1 cm at 2 m distance has been reported.

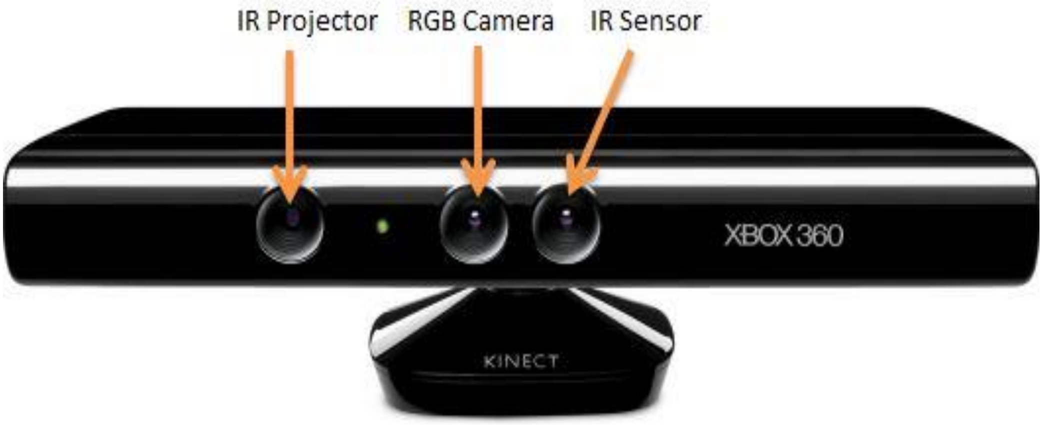


Fig. 2.13 Kinect Device: the IR projector emits a grid of IR light in front of it; this light then is reflected, by objects on its path, back to the IR sensor [12].

2.3.3 Sound

Animals and humans can determine their position and orientation in dark spaces by listening to sounds of the environment they are into. In science, this is referred to as sound localization. Sound localization is based on the propagation of sound waves in space. Positioning systems utilizing sound waves locate mobile nodes via multilateration. Static nodes are mounted on walls, ceilings or even on the floor. There are two main types of sound positioning systems, active and passive device localization. In the first type, the mobile node is a sound emitter and its location is calculated by multilateration from the fixed - already known - positions of static nodes. Alternatively, passive device localization consists of multiple static sound emitters and a passive node listener. Ultrasound is more commonly used in these systems compared to audible sound for attaining unobtrusiveness to the users. However, audible sound systems can be easily implemented through sound cards of standard devices.

Ultrasound

Ultrasounds are high frequency sound waves which are non-audible to the human ear. Distance calculation is performed through Time of Arrival (ToA) measurements. However, the operating range of ultrasound emitters is less than ten meters as a result of the sound wave intensity decay in the transmission channel. In the ToA method, synchronization of the mobile node with the static nodes is required. To overcome this problem, most ultrasound positioning systems utilize the Time Difference of Arrival (TDoA) method to enable ad-hoc localization. Mobile nodes broadcast a radio frequency signal in tandem with the ultrasound pulse to inform close-by static nodes of their presence.

Echolocation

In enclosed rooms, not only the direct sound from a sound source is arriving at the listener's ears, but also sound which has been reflected at the walls. Bats, among many other animals, use this technique called bio-sonar to determine their position and locate objects within their perimeter (Fig. 2.14). Similar solutions have been created for eliminating the need for beacons and tags in sound positioning systems. In order to determine the time periods where the direct sound prevails the secondary, object–bounced, sound, the auditory system analyzes loudness changes in different critical bands and also the stability of the perceived direction.

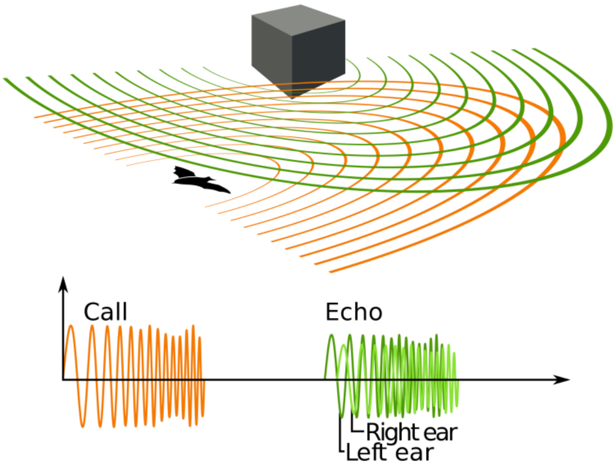


Fig. 2.14 A depiction of the ultrasound signals emitted by a bat and the echo from a nearby object [13].

2.3.4 WLAN / Wi-Fi

A Wireless Local Area Network (WLAN) is a computer network employing Radio Frequency (RF) signals for connecting several devices/computers together. Wi-Fi is the most frequently employed type of WLANs, operating based on the IEEE 802.11 standard. Nowadays, wireless network access points can be found everywhere, from commercial buildings to households. The typical coverage of a WLAN is about fifty to one hundred meters, depending on the frequency used.

WLAN positioning is a common alternative to optical positioning, since line of sight (LoS) is not required. Most WLAN localization systems are based on the RSSI (Received Signal Strength Indicators) method thanks to its ease of use. Last but not least, many standard everyday devices like mobile phones or tablets already use the Wi-Fi protocol for communication purposes, so no complementary hardware is needed for position detection. WLAN positioning combined with RSSI methods can be categorized into one of the four following approaches.

Cell of Origin (CoO)

The CoO approach is common in applications where the required accuracy level is more than fifty meters. In fact, the mobile device is assumed to have the same coordinates as the WLAN access point which emits the strongest RSSI value received by the mobile device in its vicinity. This method is adopted in positioning systems covering a fairly large area with very low accuracy requirements of the actual position of the user.

Empirical Fingerprinting

In this method, a calibration phase is initiated for accumulating RSSI values observations throughout the coverage area, uniquely marking the locations inside a building. These values are stored inside a database together with the actual building schematic, thus creating a map referred as ‘Radio Map’. When the calibration phase is complete, the mobile node compares the RSSI value measured with the database created, in order to identify its location.

WLAN Distance-Based Methods

Distance estimations can be calculated through the information contained in the arrival times and amplitudes of the received waveforms. Thus, ToA, TDoA, AoA and RTT methods can be used for distance measurement. However, WLAN standard interfaces do not provide timestamps with sufficient resolution in time and an indirect approach must be employed.

Therefore, systems using modifications in the WLAN chipset (physical layer) have been created in order to overcome this limitation of existing hardware.

2.3.5 Radio Frequency Identification

The acronym RFID stands for Radio-Frequency IDentification and refers to small electronic devices that consist of a small chip and an antenna. This small device, also referred as RFID scanner, scans its surroundings for passive or active tags attached to objects or people (Fig. 2.15). Data are reported back to the reader through electromagnetic waves. Each RFID tag has its own unique ID, typically a serial code, and other information regarding its identity. RFID tags are used in many industries (Fig. 2.16), since they can be easily attached on objects, such as clothing and pharmaceuticals, or even implanted on livestock (e.g. cattle).

Regarding positioning systems, the principles used are the CoO, RSSI and FP. The CoO method is frequently used because of its simplicity. The presence of an object, which has an RFID tag attached on, is reported back to the reader. However, the accuracy of such a system depends on the number of RFID tags used and, of course, the reading range. The RSSI method uses the range estimations between the reader and the spread tags in order to apply multilateration techniques. The Fingerprinting (FP) method on RFID positioning is applied by signal mapping of the RFID reader or the active tags. Last but not least, systems operating with the ToA, PoA and AoA methods have been introduced but have been proven difficult to be utilized due to precise time synchronization required by these methods.

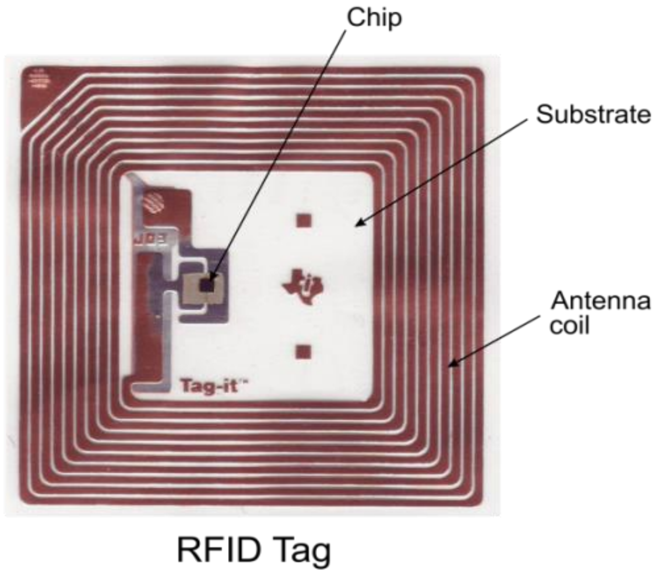


Fig. 2.15 A close look inside an RFID-tag [14].

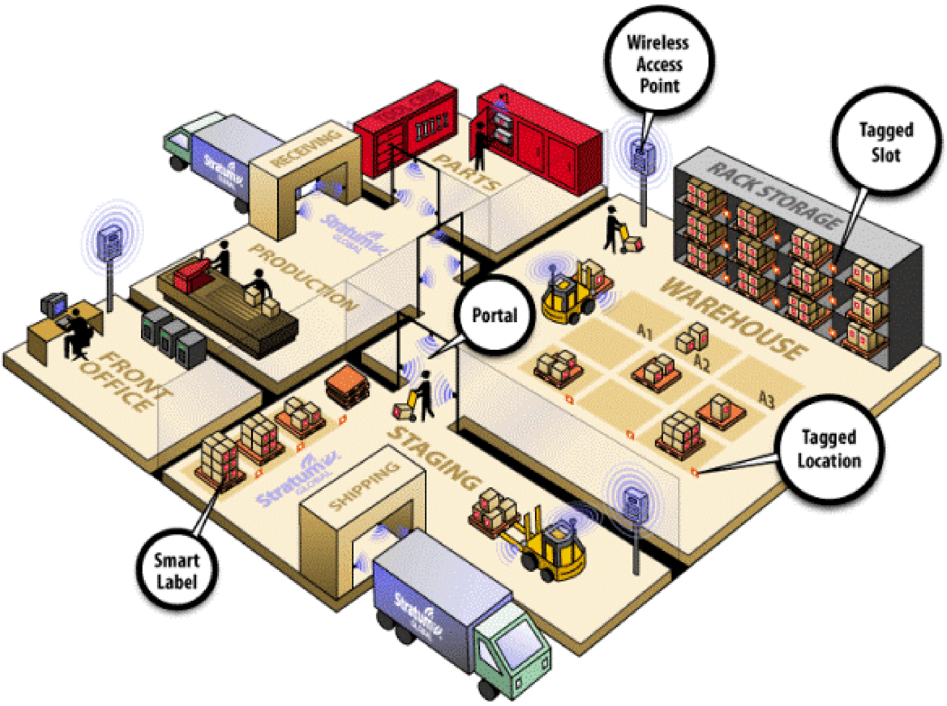


Fig. 2.16 RFID tagged industrial warehouse. Location of the working units is detected by the main server through RFID tags, spread all over the warehouse [14].

Active RFID

Active RFID systems use battery-powered RFID tags that continuously broadcast their own signal. Active RFID tags are commonly used as “beacons” to accurately track the real-time location of objects. Active tags provide a much longer reading range (above 30 meters) than passive tags, but they are also much more expensive and heavier. Positioning systems utilizing active RFID tags are based on the Fingerprinting technique with RSSI values.

Passive RFID

Passive RFID systems use tags with no internal power source and therefore do not require batteries to operate. The power required to function is transmitted from the RFID reader through radio waves (i.e. inductive coupling). When a passive tag is stimulated from the RFID reader, it reports its unique ID. Positioning systems based on passive tags consist of several tags scattered in different locations (mesh-like or grid-like). Each tag performs the role of a waypoint and their locations are stored in a database.

Passive tags are suitable for positioning purposes due to their small size, low price, cost-free installation and low maintenance needs, since no internal power source, such as batteries, is required to operate. However, a dense arrangement of tags is usually required because their detection range is often less than two meters.

2.3.6 Ultra-Wideband

Ultra-Wideband (UWB, also known as, ultra-band) is a radio technology. UWB transmits information spread over a wide bandwidth (>500 MHz) for short-ranges (i.e. less than 100 meters) with a very low energy level (Fig. 2.17). Indoor positioning utilizing UWB is a commonly used approach because of its multipath resistance and penetrability of building materials. UWB transmits in a manner that it does not interfere with conventional narrowband and carrier-wave transmission systems in the same frequency band.

The classic UWB positioning system consists of a radio wave transmitter and several receivers, which identify the broadcasted and scattered waves. A radio wave is perceived to be UWB if its bandwidth is larger than 500 MHz or 20% of the fractional bandwidth.

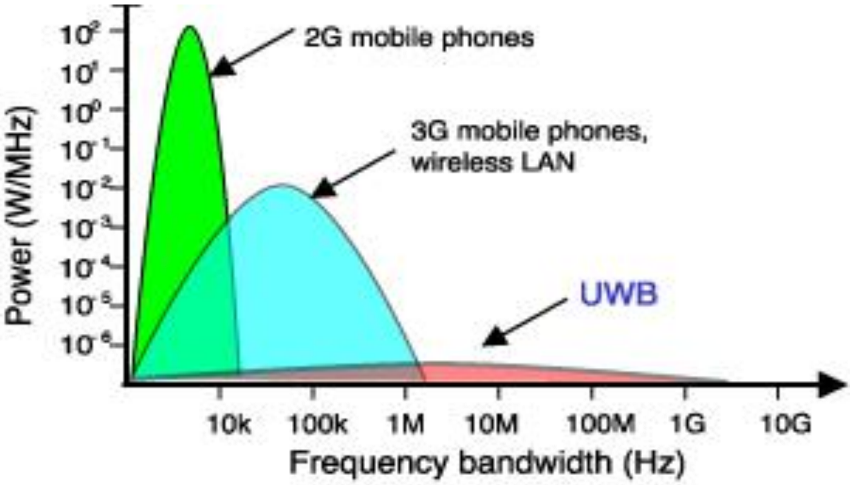


Fig. 2.17 UWB versus other radio communications systems [15].

Passive UWB localization

In passive UWB localization systems, the target's position is calculated through signal reflection (Radar principle). The typical setup consists of omnidirectional antenna emitters and listeners with known locations. Position and distance measurements of the moving target can be determined by ToA, TDoA multilateration and other range-based algorithms. In more detail, the direct (from emitters) and the reflected wave signals (from the user) are subtracted in time. Thus, their time difference reveals the distance traveled, given that the emitters and receiver antennas locations are known (Fig. 2.18).

Its main advantage is that there is no need for active or passive tag attachments. Thus, there is no restriction in the user's movement caused by wearable hardware.

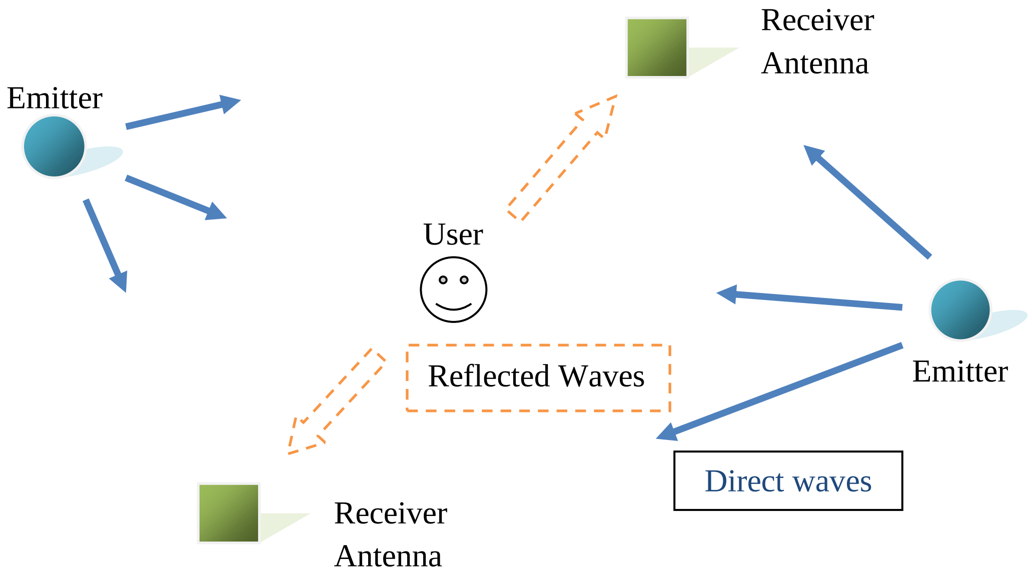


Fig. 2.18 A common Passive localization setup. Signal from emitters spread in tracking area is reflected from the user back to the receiver antennas. The difference of reflected and direct waves reveals the position of the user.

UWB Virtual Anchors

When the floor-plan information is a-priori known, the position of the receiver is estimated from the signal reflections in the room walls and the direct signal path. This means that we can acquire position information from even a single UWB transmitter. The distances derived from the delay of signal reflections from walls (virtual anchors) can be used for multilateration (Fig. 2.19).

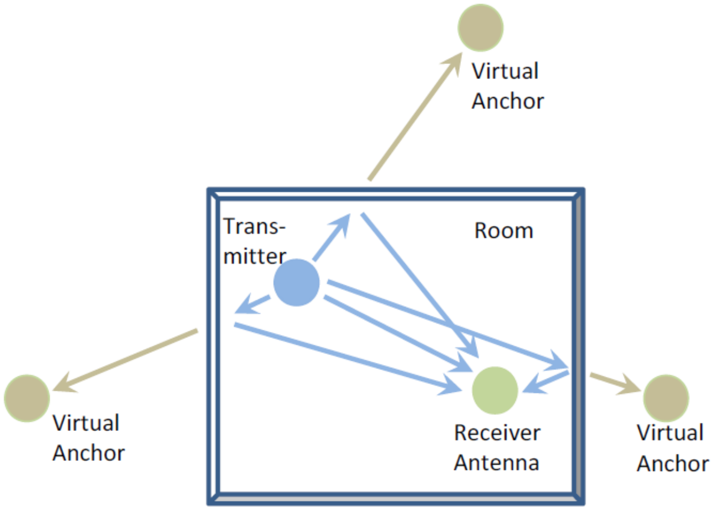


Fig. 2.19 Virtual anchor based multilateration [1].

UWB Direct Ranging

In this method, active tags are attached to the mobile node. Distance is calculated only by the direct signal path, not the reflected. As a disadvantage, LoS conditions must be met. The ToA, TDoA and RTT algorithms can be used for lateration purposes.

UWB Fingerprinting

The wide bandwidth of the channel from a UWB-based transmitter enables a fine temporal resolution of the multipath propagation model, because of the large amount of information that can be stored in it. Thus, unique location fingerprints of the positions of transmitter and receiver can be assigned in an offline phase. Its main advantage comes from the wide frequency band used in UWB. This method can achieve better accuracies compared to WLAN fingerprinting. Moreover, UWB fingerprinting can be utilized in NLoS conditions.

2.4 Summary

Regardless of the plenitude of methodologies which exist to handle indoors positioning, at the top level all technologies can be separated into three different categories based on the physical principle they utilize: inertial navigation, electromagnetic and mechanical (sound) waves. Fig. 2.20 displays a graphical overview of the existing indoor positioning technologies in terms of accuracy and coverage [1]. As can be seen from Fig. 2.21 the majority of technologies utilize electromagnetic waves. It must be noted that high accuracy systems require generally shorter wavelengths. According to these graphs, in a room-level environment with accuracy ranging from millimeters to decimeters, the most used technologies are those based on cameras, RFIDs, sound waves, UWB systems and infrared emitters. Finally, in Table 2.1 a list can be found containing all indoors positioning technologies with respect to application zone, typical coverage in meters (in terms of a single node installation), accuracy and measuring principles utilized.

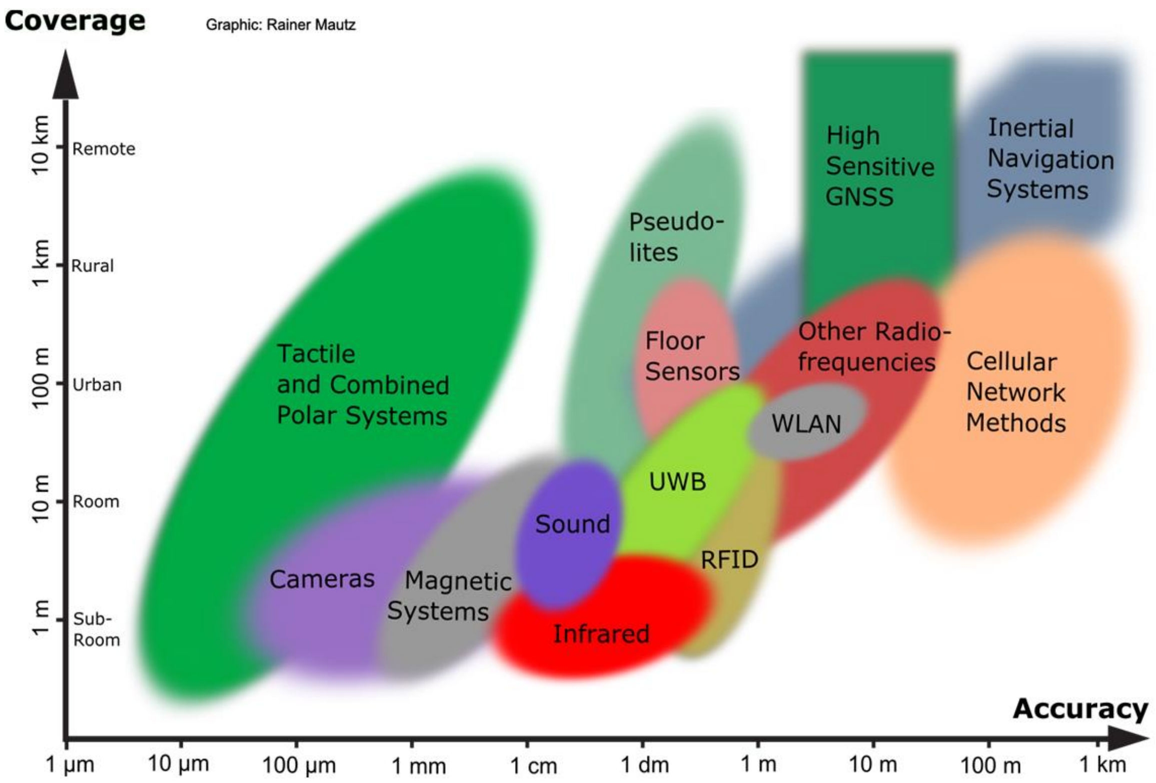


Fig. 2.20 A graphical overview of the existing indoor positioning technologies in terms of accuracy and coverage [1].

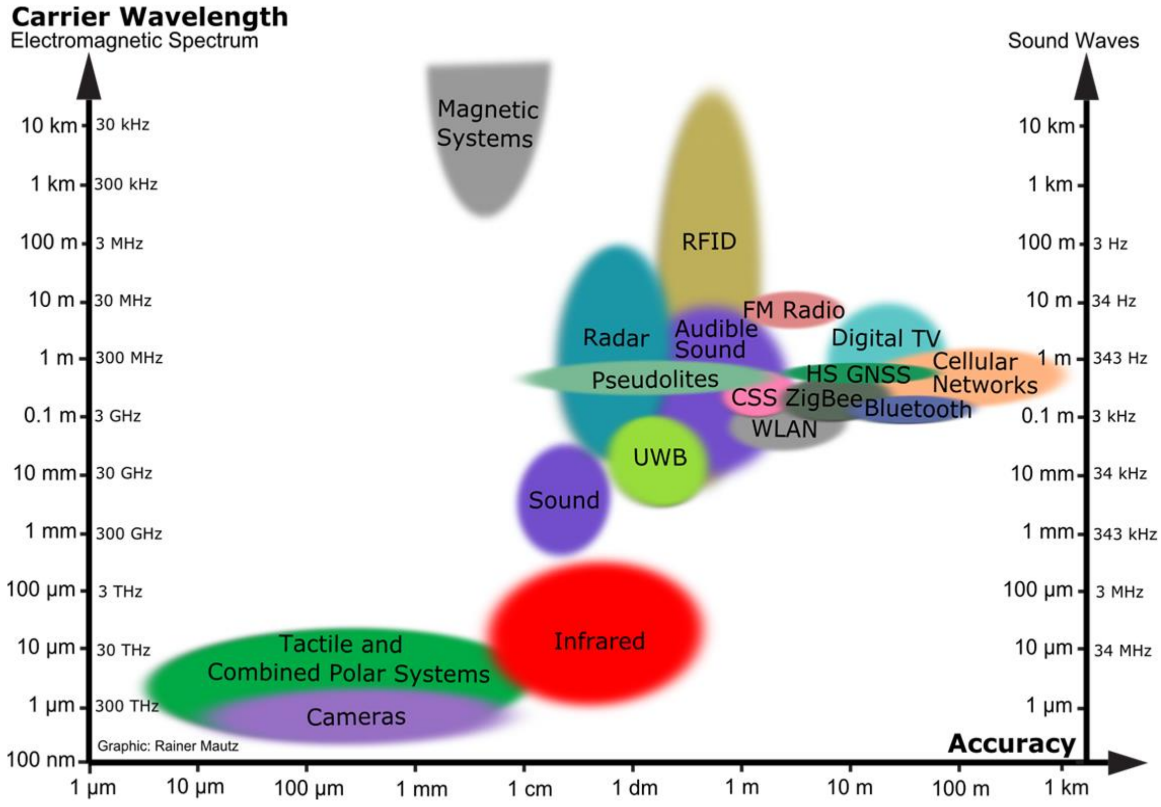


Fig. 2.21 A classification of the existing indoor positioning technologies in terms of accuracy and carrier wavelength [1].

Table 2.1 Overview of indoor positioning technologies: coverage refers to the range of a single node.

Technology	***Typical Accuracy***	***Typical Coverage (m)***	***Typical Measuring Principle***	***Main Application domain***
Cameras	0.1mm-dm	1-10	AoA	robot navigation
Infrared	cm-m	1-5	active beacons, Thermal Imaging	people tracking
Sound	cm	2-10	ToA, TDoA	collision detection
WLAN/Wi-Fi	m	20-50	FP	pedestrian navigation
RFID	dm-m	1-50	Proximity, FP	security
UWB	cm-m	1-50	ToA	automation
Inertial Navigation	1%	10-100	Dead Reckoning	pedestrian navigation
Magnetic Systems	mm-cm	1-20	FP	automotive

3. The Proposed System

The proposed approach falls into the category of ‘Imaging of Artificial Infrared Light’ in the optical-camera-based positioning systems. As mentioned in Chapter 1, in order to overcome the high cost of dedicated infrared cameras, we used two handheld console devices called Wiimotes (or Wii-remotes) that offer excellent update rate and high camera resolution. By using a Wiimote in an ‘outside-in’ manner, in the opposite way for the purpose built, we are able to track an IR light source in two dimensions. However, when tracking in two dimensions, one Wiimote is adequate since it senses the x-y plane. Object tracking in two dimensions limits us in several ways [16]. Wiimote supports a basic method to calculate depth -3rd dimension- depending on how close together are the infrared dots of the sensor bar (Fig. 3.1). Nevertheless, the blob size estimation seen from the camera is unreliable. To be able to see in three dimensions we need at least two Wiimotes.

The process of calculating depth from two or more distinct cameras is called stereo vision or imaging. If we know the location of the Wiimotes we can calculate the position based on the projections of IR blobs seen on each camera. The place where the projections intersect is the three dimensional point where the IR source was located. However, in order to be able to compute the position of a person in 3D using two Wiimotes, we need to find the rotation and the translation from one Wiimote to the other. This process is called “external calibration”. Moreover, we need to know the optical center of each Wiimote and their focal lengths. This process is called “internal calibration”. The external and internal calibration processes are essential parts of any stereo vision scenario.

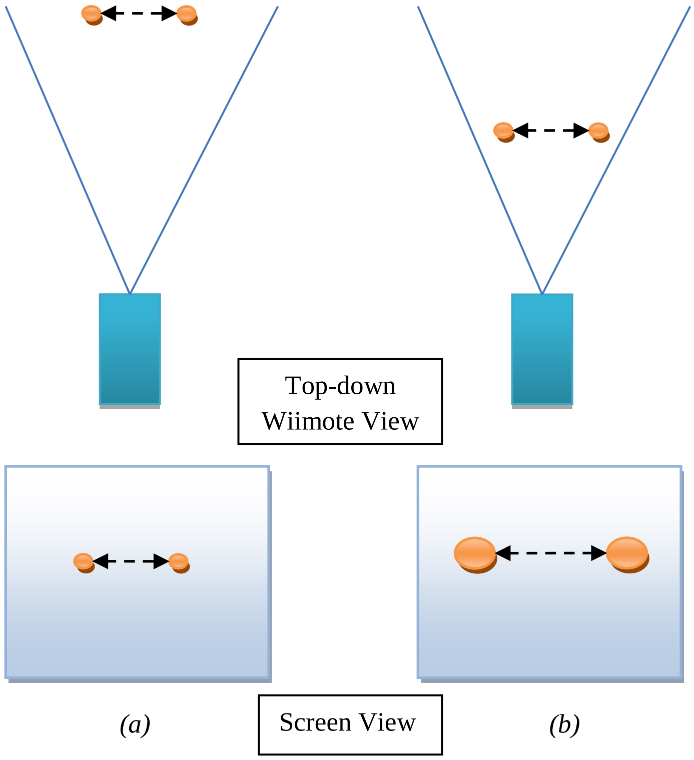


Fig. 3.1 Two infrared dots as seen from above: (a) close to the Wiimote, (b) far from the Wiimote. The distance between the two LEDs (orange circles) with respect to camera CCD resolution depends on LEDs proximity to the CCD. The depth calculation based on one Wiimote is very inaccurate.

The proposed system is based on the ‘standard lateral model’. In traditional stereo vision, standard lateral model refers to the analogy of the human eyes based vision (binocular vision). In detail, two cameras are displaced horizontally with a pre-known distance between them. Each camera obtains a different view of a scene and by comparing these two images, relative depth information can be acquired. The proposed system utilizes two Wiimotes as cameras which are connected to a standard PC through a Bluetooth communication link (Fig. 3.2). Given the distance between the two Wiimotes we are able to track the position of a self-made IR-array bulb. This bulb is mounted on top of the user being tracked. The information acquired from the two Wiimotes in the form of pixels is then transferred to the actual computer, which is responsible for calculating the coordinates of the IR-bulb in space and thus the coordinates of the user.

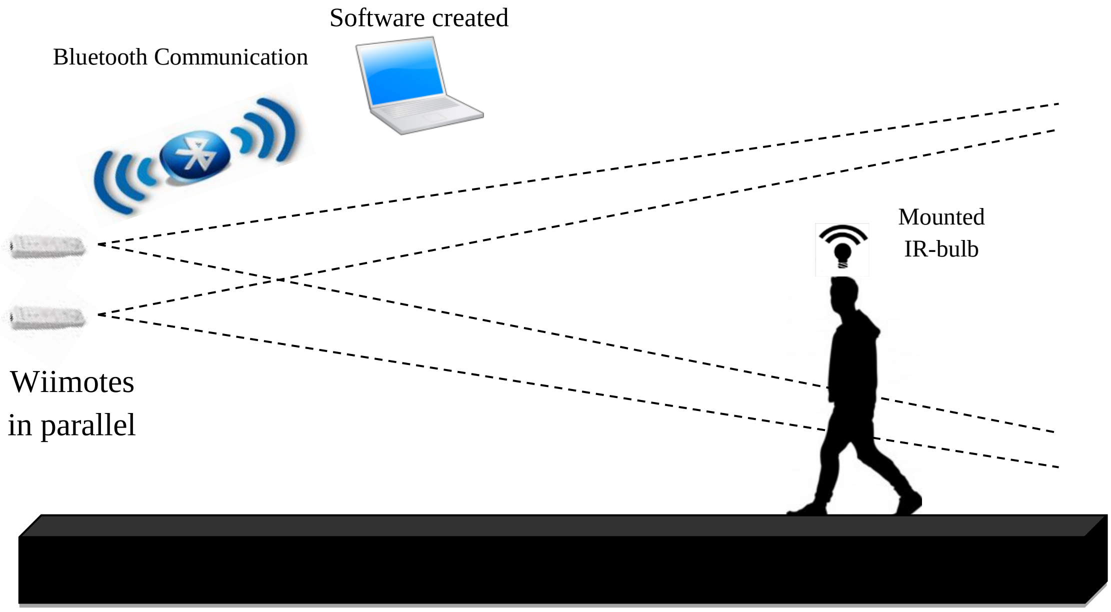


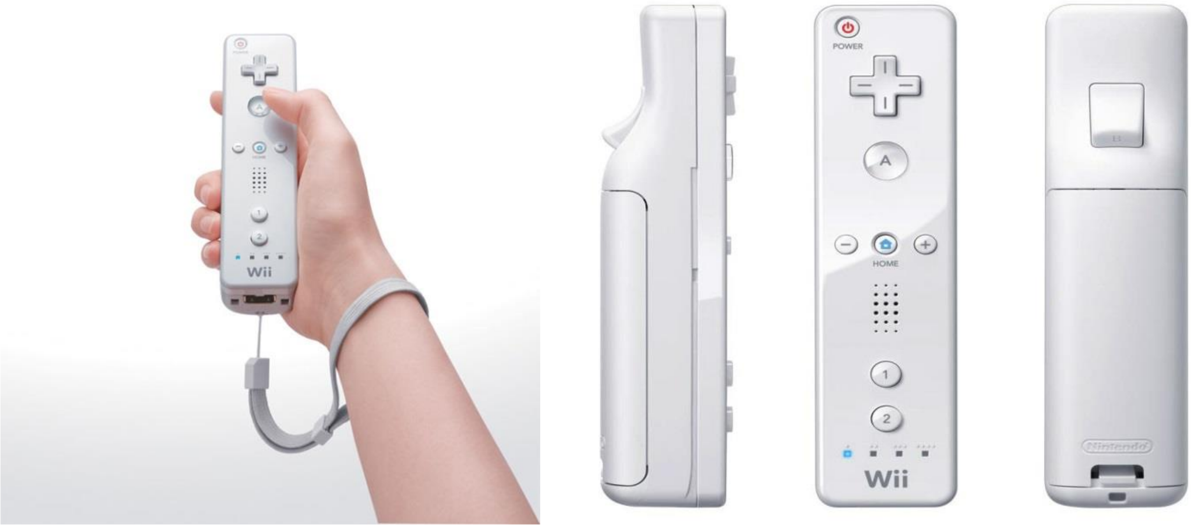
Fig. 3.2 Abstract graphical view of the proposed system: the user walking through space with the mouted IR-bulb on top of his head;the pixel coordinates are transferred through a Bluetooth link to a central PC for position calculation.

3.1 Hardware architecture

3.1.1 Wiimote specifications

The Wiimote is a wireless input device built around the Broadcom BCM2024 Bluetooth System-on-a-chip, which uses short-range Bluetooth radio to communicate with the Wii console (Fig. 3.3). It is 148 mm long, 36.2 mm wide and 30.8 mm thick [16]. The Wiimotes can operate up to 5 meters away from the Wii console. Moreover, a standard Bluetooth dongle can support four controllers paired at the same time (Fig. 3.4). The Wiimote contains multiple peripherals and an expansion port used for external add-ons (e.g. Wiimote-plus that enhances hand motion recognition), which provide data to the Broadcom chip.

Nintendo has not release official specifications regarding Wiimote internal functionality. However, a significant portion of the technical specifications has been published based on reverse-engineering techniques applied by many individuals [17].



(a) *(b)*

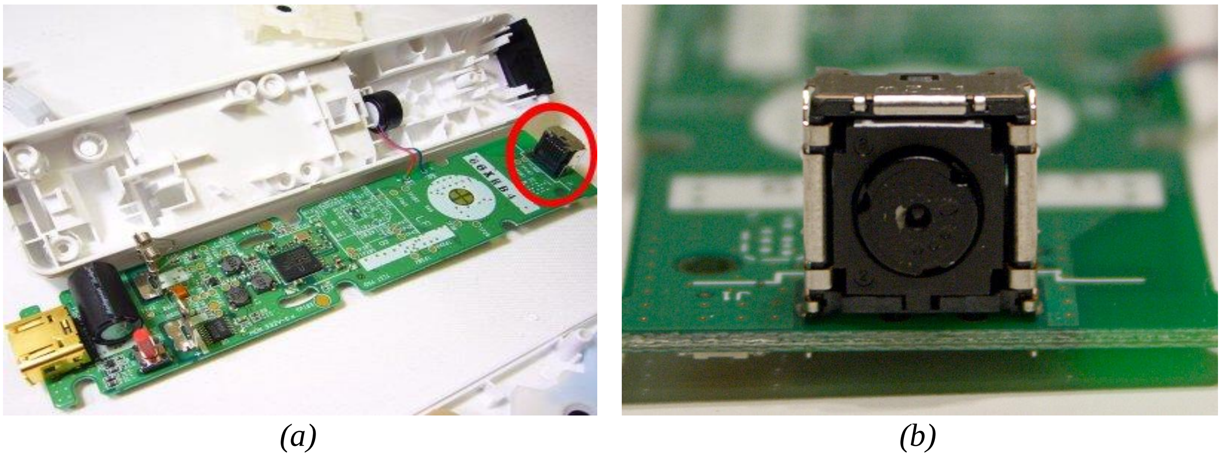
Fig. 3.3 Wiimote Controller: a) holding view and b) different angle views [18], [19].

Fig. 3.4 A family using Wiimote for home entertainment [20].

Infrared camera tracker

The infrared camera sensor is located at the tip of each Wiimote (Fig. 3.5). This camera chip is manufactured by PixArt Imaging and features an integrated multi-object tracking engine (MOT). Multi-object tracking is a computer vision technique, which enables simultaneously tracking of different objects and predicts their next move. The exact specifications of this camera remain unpublished. However, according to numerous individuals who applied reverse-engineering techniques to learn more about this sensor, it is a 128x96 monochrome camera. The actual resolution of the camera is pumped up to 1024x768 pixels due to an 8x sub-pixel analysis .This means that for each pixel in the displayed image, there are 8x more samples of pixels captured by the CCD. In addition, it provides a 100 Hz refresh rate and 4 bits of dot/blob size light intensity. The vertical field of view is 37 degrees and the horizontal is 45 degrees.

The IR-pass filter of the sensor lets 940 nm sources to be detected with approximately twice the sensitivity compared to that of 850 nm. However, it seems that the 940 nm sources are not resolved that well at close distances (<50 cm). The MOT engine implemented directly on the Wiimote also allows only the pixel coordinates of objects seen to be exported over Bluetooth connection (thus reducing the size of data to be managed), not the raw pixel data. Thus, a conventional picture cannot be captured by the Wiimote device.



(a) *(b)*

Fig. 3.5 (a) Pix-Art IR camera sensor (red circle) on top of the Broadcom BCM2042 Bluetooth System-on-chip and (b) a close view of the IR camera [21].

Accelerometer

On the top surface of the Broadcom BCM2024 board there is an ADXL330 three-axis linear accelerometer. This accelerometer is responsible for the Wiimote motion sensing. It features a +/−3 g sensitivity range with 8 bits per axis, as well as a 100 Hz update rate.

Bluetooth connectivity

The Broadcom 2024 chip used by Wiimote is designed for computer mice and keyboards and complies with the Bluetooth Human Interface Device (HID) standard. As such, the Wiimote utilizes the standard Bluetooth HID protocol. This protocol is based upon the USB

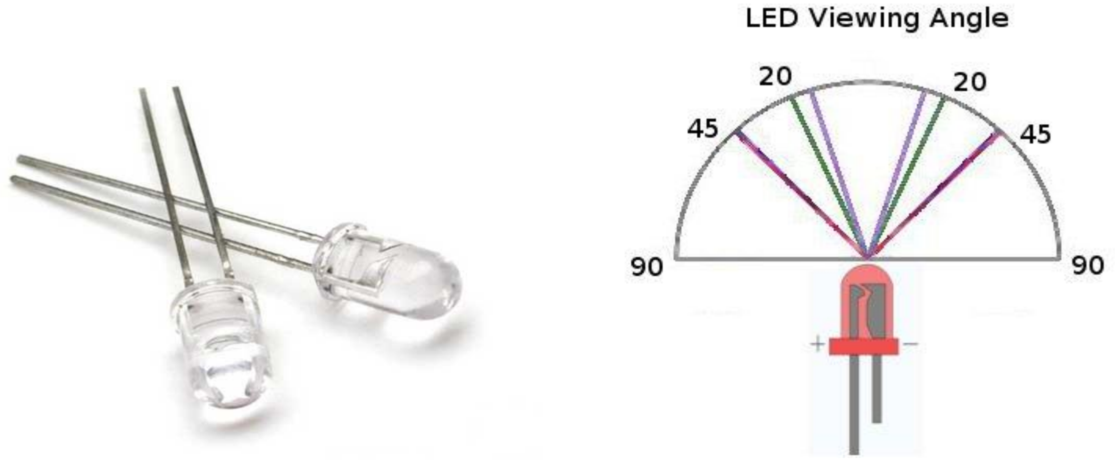
HID standard. However, the Wiimote does not make use of the standard data types and HID descriptor, and only describes its report format length, leaving the actual content undefined, which makes it useless to standard HID drivers. The HID output reports of the Wiimote contain a rather complicated set of operations and its HID input reports return a number of different data packets containing data from its peripherals. Although Wiimote will appear as a standard input device to any Bluetooth host, it is not one hundred percent compliant with the HID standard.

Energy consumption

The Wiimote uses two AA batteries and has an operating time between twenty and forty hours, depending on the number of active components. Approximately eight bits of battery-voltage resolution are available.

3.1.2 Hardware Components

In addition to the two Wiimotes, an IR-bulb array is also used in the proposed positioning system, which is mounted on top of the user. This bulb is actually a white plastic ball with an array of six 850 nm IR LEDs inside. The reason behind this is that a single common IR-LED is highly one-way directional with about fifteen to twenty degrees angle beam (Fig. 3.6). To overcome this problem, the white plastic ball was used to scatter the IR light across the sides of the ball.



(a) *(b)*

Fig. 3.6 (a) the 940 nm IR-LED used in the proposed system, (b) a plot of the IR-LED emission angles [22].

According to the manufacturer specifications, the current required to power a single IR LED is 20 mA with voltage drop of 1.2 V. To supply the six LEDs in series that were employed in the proposed system, a 9 V battery with a 100 Ohm resistor were used.

For the Bluetooth connection a generic MSI Bluetooth dongle was used. A list with all compatible Bluetooth dongles devices can be found in [23].

3.1.3 Unused Hardware

The Wiimote also contains several hardware parts that were not used in this project. They are summed up here so that it is clear that the Wiimote perform additional operations to those described in this thesis. Wiimote contains a “Rumble” module which lets the Wiimote vibrate to notify the user of in-game events, a speaker for playing sounds and an expansion port for connecting several expansions to the Wiimote, such as the “motion plus +” which enables more complex hand movements to be recognized.

3.2 General camera mathematics

The pinhole camera model describes the simplest optical device that shapes an image onto its image (or focal) plane, without the need of lenses or mirror for focusing light [24]. The image plane is the plane where the image is formed after the light passes through the camera hole. The distance between the camera hole and the image plane is referred to as “focal length”. This model demonstrates the mathematical relation between a point in the 3D space and its projection onto the image plane.

Based on the pinhole camera model, a simple camera captures the image of the area under monitoring through its lens and the inverted image is recognized by the CCD sensor of the camera (Fig. 3.7). The inverted image plane is by definition the virtual surface in the back of the camera, containing the inverted image captured by it.

Assuming we monitor a tree in a certain depth, the following scenario takes place (Fig. 3.8).

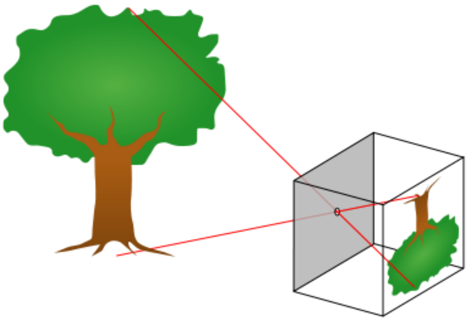


Fig. 3.7 Pinhole Camera model: a simple camera captures the image of the area under monitoring through its lens and the inverted image is recognized by the CCD sensor of the camera [25].

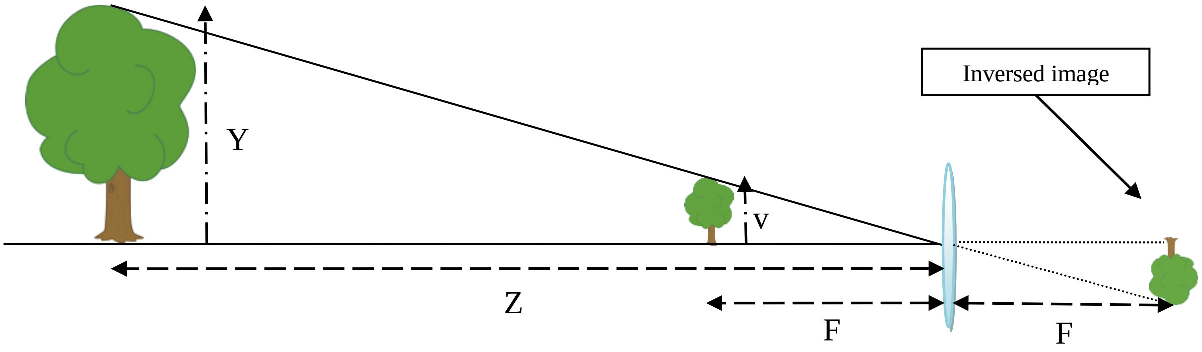


Fig. 3.8 A tree being captured through the camera lens; if we rotate the reversed image plane by 180^o in the depth axis the resulting image plane gets in front of the aperture and is separated by the distance of the camera focal length.

The parameter F in Fig. 3.8 denotes the focal length of the camera. The focal length of an optical system is a measure of how strongly the system converges or diverges light. Moreover, focal length is the distance between the inverted image plane, containing the captured image, and the aperture. As can be seen in Fig. 3.8, if we rotate the reversed image plane by 180^o in the depth axis, then the resulting image plane gets in front of the aperture and is separated by the distance of the camera focal length. This generates a virtual front image plane which does not violate the mathematical model of the pinhole camera and is simpler to analyze. Parameter Y denotes the actual height of the tree and Z the actual depth. Parameter v is the height of the tree into the captured 2D scene.

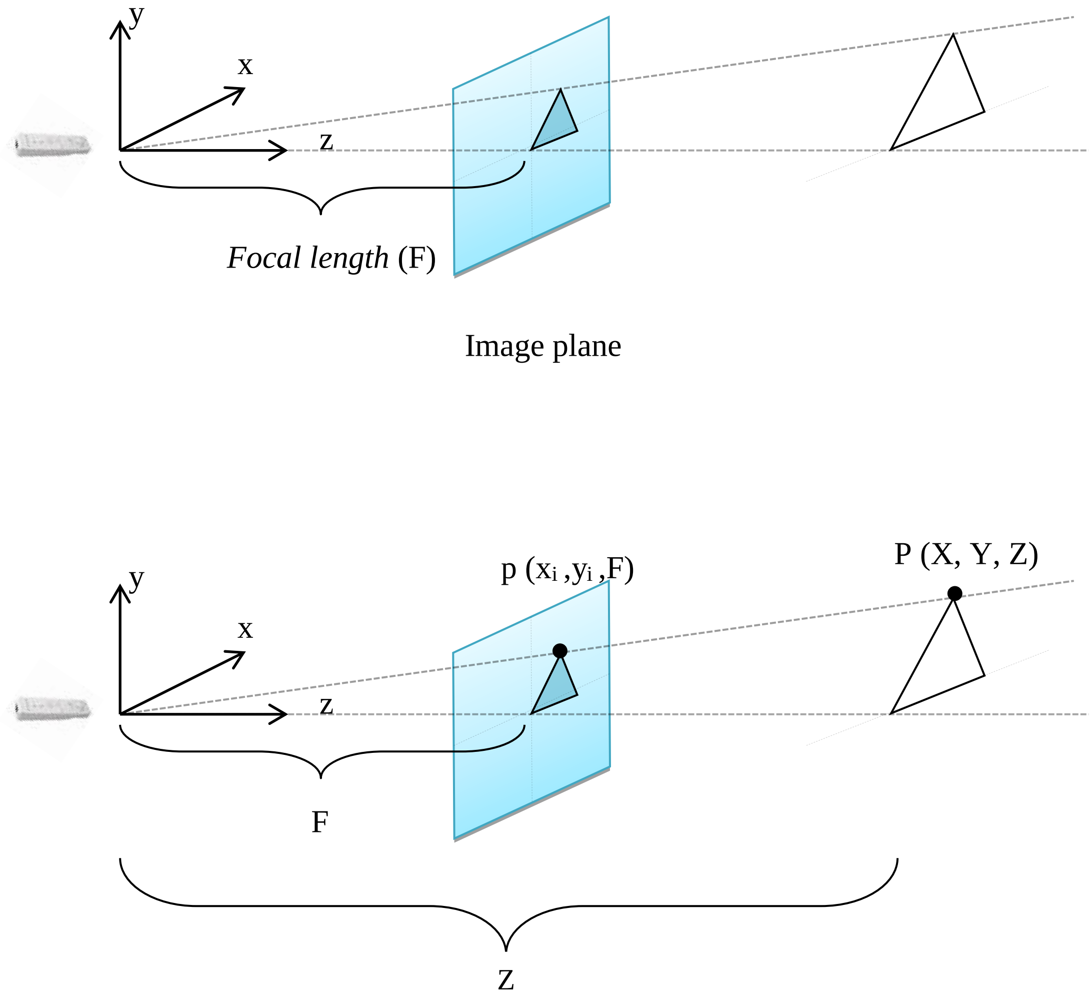


Fig. 3.9 A *camera capturing an abstract triangle shape object through its lens: parameter F denotes the focal length of the camera; point P(X, Y, Z) is the top point of the vertex of the triangle shown in the right, point p ($x_{i,}$ y_i, F) is the top point of the vertex of the small triangle formed in the image plane; the origin of the x-y-z axes corresponds to the CCD sensor of the camera.*

Assuming we have a camera looking at a triangle (Fig. 3.9) and our center of coordinates is the CCD sensor, then given two points *p* and *P*, their coordinates will be *p*(x_i ,y_i ,*F*) (where x_i is the distance from the axis x into the image plane, y_i the distance according to the y axis and *F* is the focal length of the camera, all in the same units) and *P(X ,Y ,Z)* (where *X, Y, Z* are the actual coordinates of the edge of the big triangle), respectively.

According to the rule of similar triangles, the following equation applies:

$$\frac{x_i}{X} = \frac{y_i}{Y} = \frac{F}{Z} \tag{3.1}$$

From the above equation we can get image plane coordinates by solving for *x_i* and *y_i*, as follows:

$$x_i = F\frac{X}{Z} \tag{3.2}$$

$$y_i = F\frac{Y}{Z} \tag{3.3}$$

The above relations are derived based on the assumption that the cameras can be approximated by the pinhole camera model.

However, since we acquire from the CCD sensor the image plane points in pixels, the next step is to transform the units (lengths) into pixels. Given a camera resolution (e.g. 1024x768) we denote by c_x and c_y the center of x and y axis, respectively, in the image plane (Fig. 3.10).

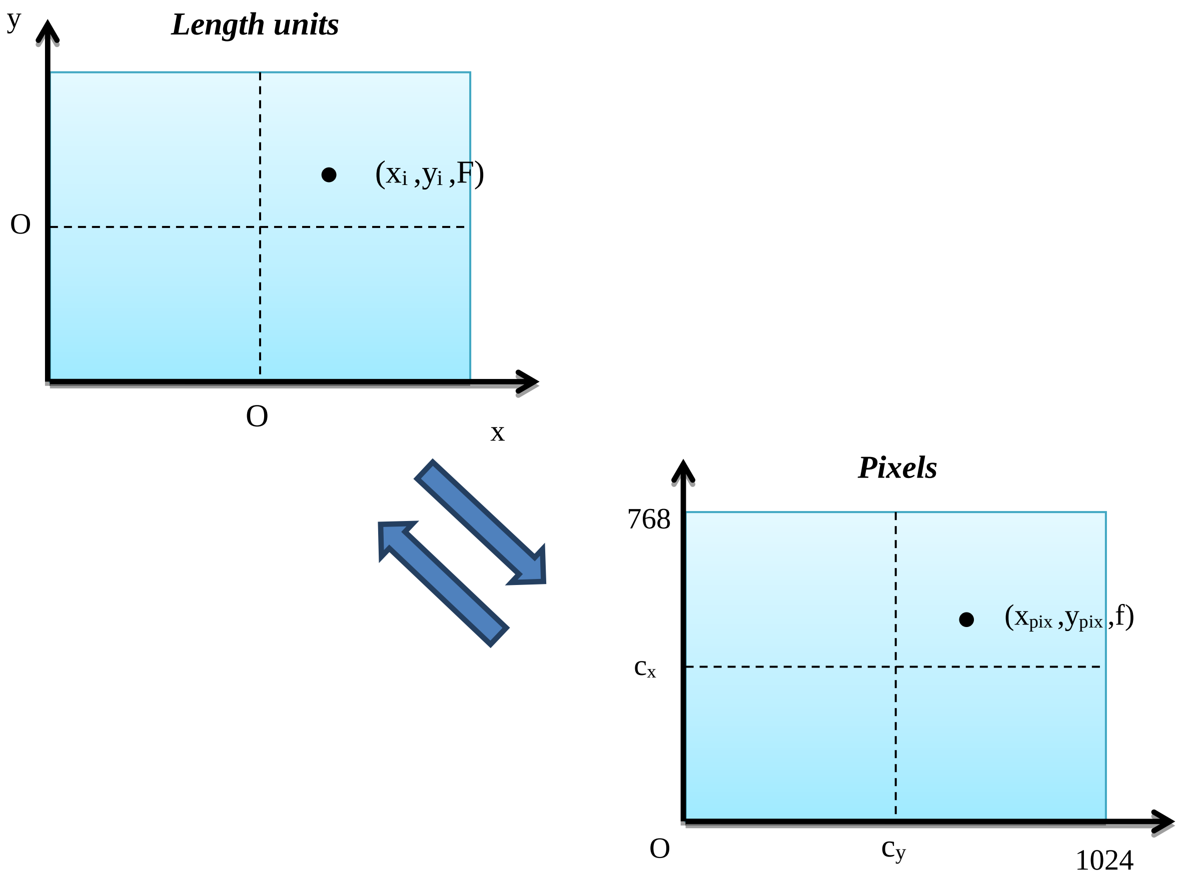


Fig. 3.10 The image planes measured in length units (e.g. cm, mm etc.) and pixels, respectively and the conversion from pixels to length units and vice versa.

From Fig. 3.10 it is derived that:

$$x_{pix} = x_i * \frac{1}{kx} + c_x \tag{3.4}$$

$$y_{pix} = y_i * \frac{1}{ky} + c_y \tag{3.5}$$

where:

x_{pix}, y_{pix} - the position of a random point on the image plane, measured in pixels

x_i, y_i - the position of a random point on the image plane measured in length units (e.g. cm, mm etc.)

kx, ky - scaling factors that corresponds to the portion of units per pixel

c_x, c_y - the center of x and y axis, respectively, in pixels (e.g. 512, 384)

By combining (3.4), (3.5) with (3.2), (3.3), the following equations are derived:

$$x_{pix} = x_i * \frac{1}{kx} + c_x = \frac{Fx}{kx}\left(\frac{X}{Z}\right) + c_x = f_x\left(\frac{X}{Z}\right) + c_x \tag{3.6}$$

$$y_{pix} = y_i * \frac{1}{ky} + c_y = \frac{Fy}{ky}\left(\frac{Y}{Z}\right) + c_y = f_y\left(\frac{Y}{Z}\right) + c_y \tag{3.7}$$

where:

Fx, Fy -are the focal lengths of the camera given in length units (e.g. cm, mm etc.).

f_x, f_y -are the focal lengths of the camera given in pixels, according to the following equations:

$$f_x = \frac{Fx}{kx}, f_y = \frac{Fy}{ky} \tag{3.8}$$

Although in an ideal pinhole camera, the focal lengths of x-axis is equal to that of the y-axis $(f_x = f_y = f)$, in real cameras they can differ by a small amount, thus the different notation. The reasons they differ are the following:

- flaws of the digital camera sensor
- the image has been non-uniformly scaled during post-processing
- the camera lens introduce unintentional distortion
- the camera uses an anamorphic format, where the lens compress a widescreen scene into a standard-sized sensor
- errors in camera calibration

In all of the above cases, the resulting image has non-ideal, non-square pixels. Following from this, there is a difference between the focal length of x and y axis, respectively.

Summarizing all the above into one useful equation:

$$Z * \begin{bmatrix} x_{pix} \\ y_{pix} \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{3.9}$$

Equation (3.9) denotes that the 2D image point in pixels equals a matrix containing the intrinsic camera parameters multiplied by the actual 3D point coordinates in units. Parameter Z on the left side of the equation (3.9) is a scaling factor and can be used to change the type of unit measuring (e.g. from cm to mm etc.).The matrix shown above is called calibration or intrinsic matrix. The intrinsic matrix is parameterized according to R. I. Hartley, A. Zisserman [24] as follows:

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3.10}$$

Parameter S in equation (3.10) denotes the axis skew. Axis skew causes shear distortion in the projected image. Furthermore, we can decompose the intrinsic matrix into a sequence of shear, scaling, and translation transformations:

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & c_x \\ 0 & 1 & c_y \\ 0 & 0 & 1 \end{bmatrix}}_{2D\ Translation} * \underbrace{\begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{2D\ Scaling} * \underbrace{\begin{bmatrix} 1 & s/f_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{2D\ shear} \tag{3.11}$$

3.3 Internal Calibration

Internal calibration is the task of associating the ideal pinhole model with an actual camera device. This calibration defines the camera properties like focal length and shear distortion of camera image. In ordinary cameras, the chessboard method described in [26] is the prevailing technique for internal calibration. It involves taking several screenshots of a regular chessboard with known distances between the chess squares from different angles and pre-known distances for measuring the camera properties. However, in our case, since the

Wiimote device cannot output actual camera images, it is difficult to utilize this technique. Consequently, we approximated the internal parameters utilizing the measured Field of View (FOV).

For simplicity reasons, we assume as optical center the center of the visible field, given the camera analysis (1024x768) at 512,384 pixels. No pincushion or barrel distortion in the camera image has been taken into account. In general, focal lengths of different axis f_y , f_x are similar. In our case we assume the same. Below, we present the mathematical equation used for calculating the focal length of one Wiimote in pixels:

$$\mathrm{f} = \frac{0.5\text{*"horizontal resolution"}}{\tan(0.5\text{*}\frac{\pi\text{*"horizontal field of view"}}{180})} \tag{3.12}$$

When we insert the values described in paragraph 3.1.1, the estimated internal matrix for a Wiimote is approximately given by the following matrix:

$$K = \begin{bmatrix} 1236.1 & 0 & 512 \\ 0 & 1236.1 & 384 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.13}$$

The axis skew (s) is considered to be equal to zero.

3.4 External Calibration

To compute the position of the person in the 3D space using two cameras, we need to find the rotation and the translation from one camera to the other (Fig. 3.11). This comprises the external calibration process. External calibration gives us the extrinsic matrix, which contains the rotation and translation mentioned above. In our case, the origin of the world coordinates and the origin of the left-first camera CCD coordinates are the same. Moreover, we used the cameras in parallel, with a known distance (disparity), which in stereo vision is known as standard lateral model. Thus, given the translation vector T (3x1) and the rotation matrix R (3x3) describing the transformation from the left camera to the right camera coordinates, the equation to solve for stereo triangulation is the following:

$$p' = RK\,p + T \tag{3.14}$$

where:

p'(3x1) is the observed point p, translated and rotated into world coordinates,
K (3x3) is the internal matrix,
R (3x3) is the rotation matrix and
T (3x1) is the translation vector.

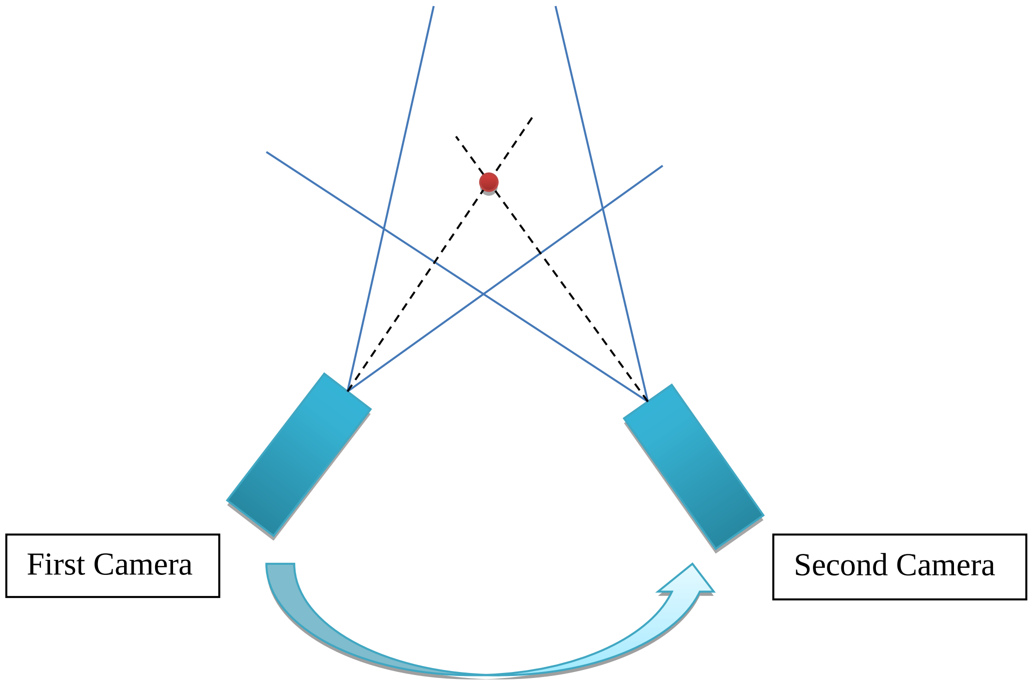


Fig. 3.11 Visual concept of external calibration from one camera to the other. External calibration creates the external matrix containing the rotation and translation from the first to the second camera CCD coordinates.

3.5 Tracking

In order to track the movement of a person in a room-level environment, we acquired first two Wiimotes and placed them parallel to each other, with a known distance apart. The world coordinates are the same with the camera coordinates and the origin point is the CCD sensor of the left Wiimote. We moved the second Wiimote by twenty centimeters in the x-axis keeping the Wiimotes in parallel. The distance between the CCD sensors denotes the Baseline as mentioned in the bibliography [24]. Then, a triangle is formed by the observed point (person under monitoring) and the known coordinates of the CCD sensors (Fig. 3.12). To solve the triangulation problem we need to know the sides of the triangle, however that is not possible. Keeping in mind that this is not a 2D but a 3D problem, we approached it in the following way.

Assuming point A1 is the origin of the world coordinates, A1 (0, 0, 0) and B1 is the second Wiimote sensor coordinates, B1 (20, 0, 0) as shown in Fig. 3.12, we denote as A2 the point created from the intersection of the ray which connects the first camera CCD and the observed point P(X, Y, Z) with the first camera's image plane. Likewise, B2 is denoted as the point created by the intersection of the second ray which connects the second camera CCD and the observed point P(X, Y, Z) with the image plane of the second camera. At this point we should present a third dimension to our 2D points, thus we apply the focal length of Wiimotes as a third dimension and reforming them as A2(x1, y1, F) and B2(x2, y2, F).

First, we need the projection coordinates of the image planes A2, B2 in pixels. To get the actual coordinates according to world reference point we must convert them to units and then rotate and translate them, according to the following equations:

$$A2 = inverse(K1) * (R1 * projection1) + T1 \tag{3.15}$$

$$B2 = inverse(K2) * (R2 * projection2) + T2 \tag{3.16}$$

where:

$K1, K2$ (3x3) - the internal Wiimote matrices

$R1, R2$ (3x3) - the rotation matrices

$projection1, projection2$ (3x1) - pixel coordinates on the image plane

$T1, T2$ (3x1) - the translation vectors

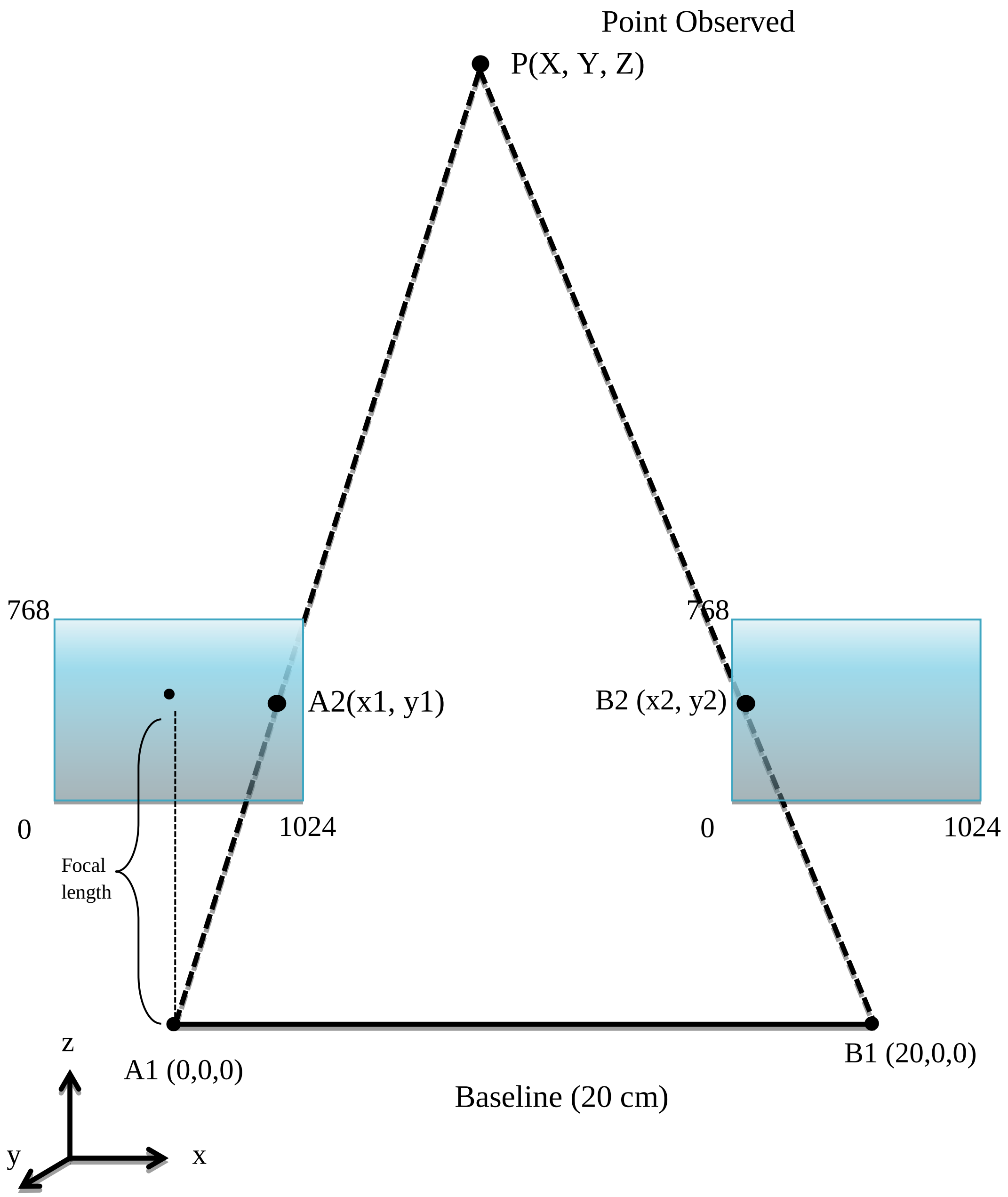


Fig. 3.12 Point observation through each Wiimote image plane: points A2, B2 are the intersections of the ray meeting the CCD sensors at point A1, B1 respectively. A1 represents the first camera's CCD which is also the reference point of our world coordinate system.

Now, we are able to find the equations of the 3D lines created between A1-A2 and B1-B2. A line in the form of vectors in the 3D space (Fig. 3.13) between two points $P(x, y, z)$and $P_0(x_0, y_0, z_0)$ is denoted as follows:

$$\tilde{r} = \tilde{r_0} + \tilde{a} = \tilde{r_0} + t * \tilde{v} \qquad (3.17)$$

where:

$\tilde{r}$ - is the vector of point $P(x, y, z)$
$\tilde{r_0}$ - is the vector of point $P_0(x_0, y_0, z_0)$
t - is a number which multiplies vector $\tilde{v}$
$\tilde{v}$ - is a vector parallel to the line

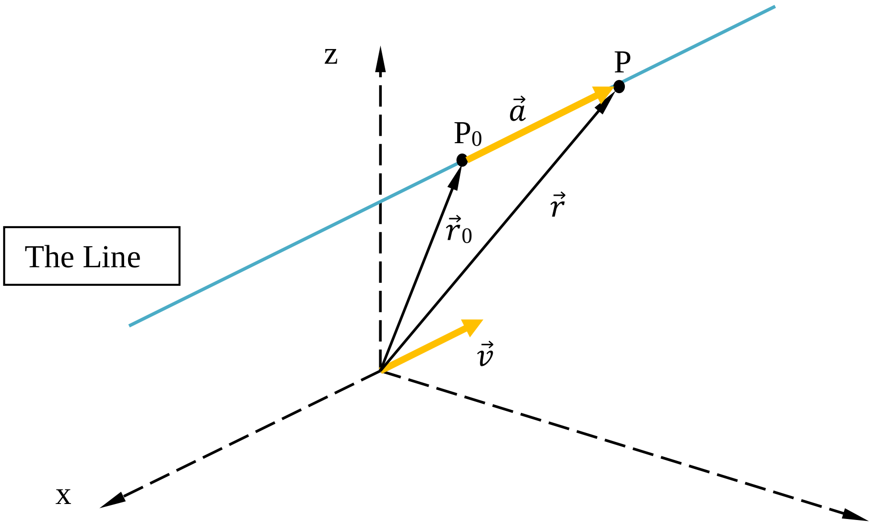


Fig. 3.13 A line in vector form in the 3D space between two points P, P₀ [27].

The line defined by A1-A2 as a function of parameter t, is given by:

$$A(t) = A1 + t * (A2 - A1) = A1(1 - t) + A2 * t \qquad (3.18)$$

The line defined by B1-B2 as a function of parameter s, is given by:

$$B(s) = B1 + s * (B2 - B1) = B1(1 - s) + B2 * s \qquad (3.19)$$

The next step is to solve the system between the two lines created. However, in 3D space there is generally no solution given from solving a formal line intersection problem like in 2D because the lines might probably not intersect. The best solution in such problems is given by linear algebra considering finding the point of closest approach. To find the closest point of approach between the two lines in 3D space, we minimize the difference A(t)-B(s) taken in each dimension. Since A1, A2, B1, B2 are all row vectors, the minimization product in Matlab form (for simplicity reasons) is given by:

$$ts = \frac{[A2(:) - A1(:), -(B2(:) - B1(:))]}{(B1(:) - A1(:))} \qquad (3.20)$$

More details on the closest point of approach can be found in [28], [29].

In order to calculate the closest point of approach/intersection we solve one of the two line equations (3.18) or (3.19) with the minimization product ts (3.20). The vector formed from each solution is our observed point.

$$A(ts) = A1 + ts * (A2 - A1) \tag{3.21}$$

$$B(ts) = B1 + ts * (B2 - B1) \tag{3.22}$$

A(ts) and B(ts) vectors given from (3.21) and (3.22) respectively, represent the closest point of approach between the two lines in the 3D space. It must be mentioned that the method described above applies well on any pair of lines and in any number of dimensions as long as the lines are not parallel, in which case there are infinitely possible pairs for A(t) and B(s).

3.6 Tracking Volume

A simple, yet effective, way to calculate the tracking volume of the proposed system is through geometry. One Wiimote can track an IR light source up to five meters away. However, the viewing angle is quite narrow; at about 45^{o} degrees on horizontal axis and 35^{o} on the vertical axis. Nevertheless, the tracking volume of one Wiimote can be drawn as a cone, like in any camera’s field of view (Fig. 3.14).

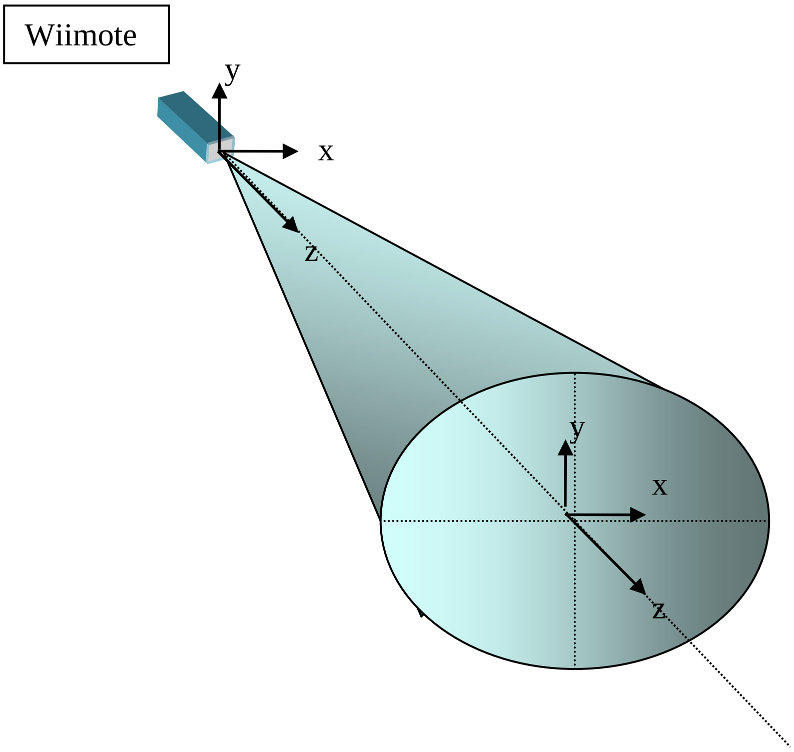


Fig. 3.14 The view cone of one Wiimote in 3d representation. X-axis refers to the horizontal axis, Y-axis on the vertical axis and Z-axis is the depth axis.

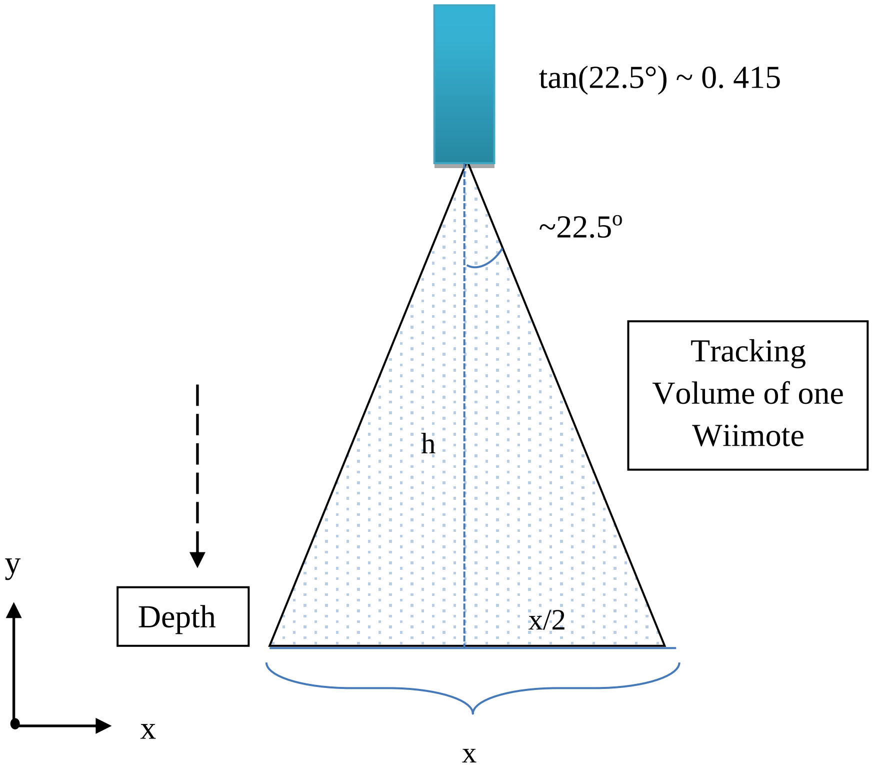


Fig. 3.15 The tracking cone area of one Wiimote as seen from above in relation to its depth view. The isosceles triangle formed is the tracking area.

According to Fig. 3.15, the depth can be derived as follows:

$$h = \frac{x/2}{\tan(22.5°)} \tag{3.23}$$

When the stereo vision scenario takes place by using two Wiimotes in parallel, an isosceles triangle is shaped that forms the tracking volume. The magnitude of this triangle depends on the distance of the Wiimotes.

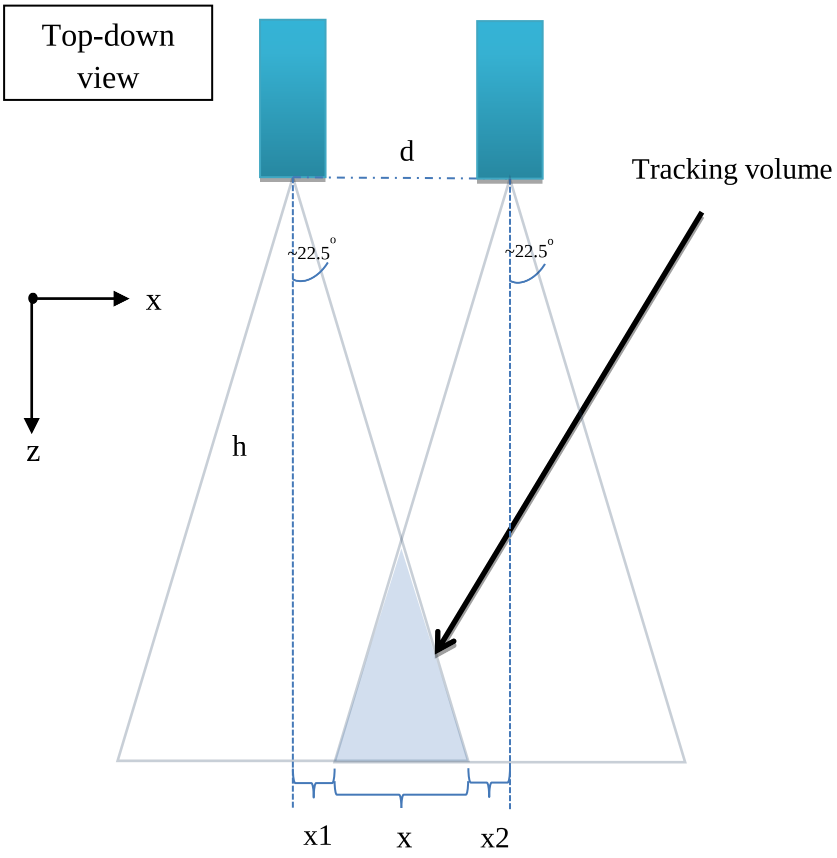


Fig. 3.16 The actual tracking volume of the proposed system as seen from above; an isosceles triangle is formed (blue triangle).

Given that the two Wiimotes are in parallel, the two isosceles triangles formed have the same dimensions (Fig. 3.16), thus it results that:

$$x1 = x2 \tag{3.24}$$

$$d = x1 + x2 + x \tag{3.25}$$

However, it holds that:

$$x1 + x = h * \tan(22.5°) \tag{3.26}$$

$$x2 + x = h * \tan(22.5°) \tag{3.27}$$

By adding equations (3.26) and (3.27) it results that:

$$x1 + x2 + 2 * x = 2h * \tan(22.5°) \tag{3.28}$$

Solving (3.28) for x and considering (3.25), results in the following equation:

$$x = 2h * \tan(22.5°) - d \tag{3.29}$$

Parameter x represents the base of the triangle which forms the tracking space of our system in the x-axis.

For the z-axis (i.e. depth axis), we need to calculate first the angle φ of the isosceles triangle formed. Angle φ is equal with 22.5^{o} since it is an alternate exterior angle, thus it results that:

$$z = \frac{x/2}{\tan(22.5°)} \tag{3.30}$$

By using equation (3.29) for x, it results that:

$$z = h - \frac{d/2}{\tan(22.5°)} \tag{3.31}$$

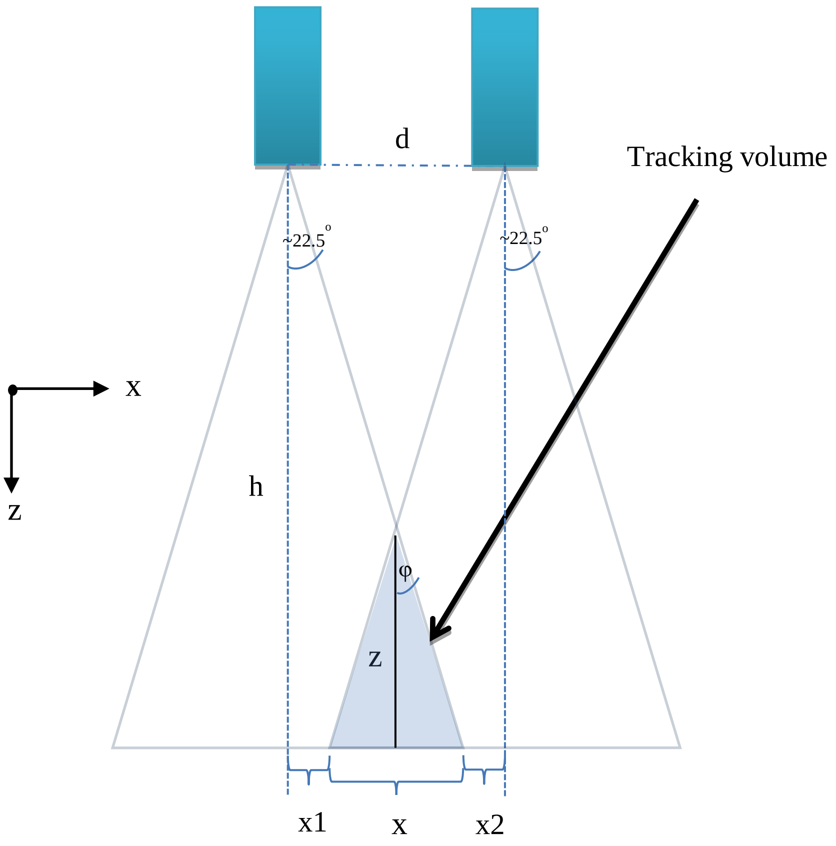


Fig. 3.17 Calculation of parameter z which denotes the depth of the tracking volume based on the distance between the Wiimotes.

Given that the two Wiimotes are in parallel, the height (y-axis) of the tracking area is proportional to the depth (z-axis) and can be easily calculated as follows:

$$y = 2h * \tan(17.5°) \qquad (3.32)$$

3.7 Software architecture

3.7.1 Libraries

As mentioned in paragraph 3.1, the Wiimote will operate as a Bluetooth HID device (controller) on any host computer. However, a driver should be installed on that host computer in order to be able to operate properly.

Software libraries for connecting a Wiimote to a PC, which translate the input report data and analyze the returned data, are available for nearly every major development platform on

Windows, Mac OS, and Linux. After a thorough research, it was decided to utilize the Wiiuse library. Wiiuse is a library written in the C programming language, which supports motion sensing, IR tracking and many more operations. It is single threaded and non-blocking, thus comprising a light weight and clean API. The version deployed in the present thesis as an external library is Wiiuse v0.12 and is distributed under the GPL 3+ non-commercial use. Wiiuse offers a variety of functions for working with the infrared camera which is the target of our application. After downloading the source code from [30] we compiled the code as a dynamic library and linked it to our application.

3.7.2 Qt framework

The actual application was implemented in Qt-Creator; the IDE of Qt. Qt is a cross-platform application framework, which is used mainly for application software development. This framework is capable of running on different software platforms without altering the entire coding structure. In addition, Qt offers a wide variety of graphical widgets without the need of using OpenGL for rendering 2D and 3D vector graphics. Qt uses standard C++ with extensions including signals and slots that simplify handling of events. Qt was an excellent choice for developing a Graphical User Interface (GUI) in tandem with our main application for observing the position of IR stimulus on the computer screen.

3.7.3 Implementation

The application created is called “WiiIRLocator”. The User Interface (UI) consists of several buttons and indications concerning the actual use of Wiimotes. Moreover, there is a graphics view widget that is capable of showing the IR stimulus to the user. More specifically, when the user starts the application, she/he will be prompt to connect the Wiimotes, given that she/he has already paired them through Bluetooth connectivity in her/his computer. Pairing is a process enabling communication between two devices using Bluetooth radio waves, by agreeing on a ‘Bluetooth profile’ which is responsible for the handshaking process between the two devices. The detailed process of pairing Wiimote devices can be found in [31]. For our case, two Wiimotes should be connected, in order to calculate her/his position in space. However, the infrastructure for more Wiimotes has also been developed as an extension for future applications.

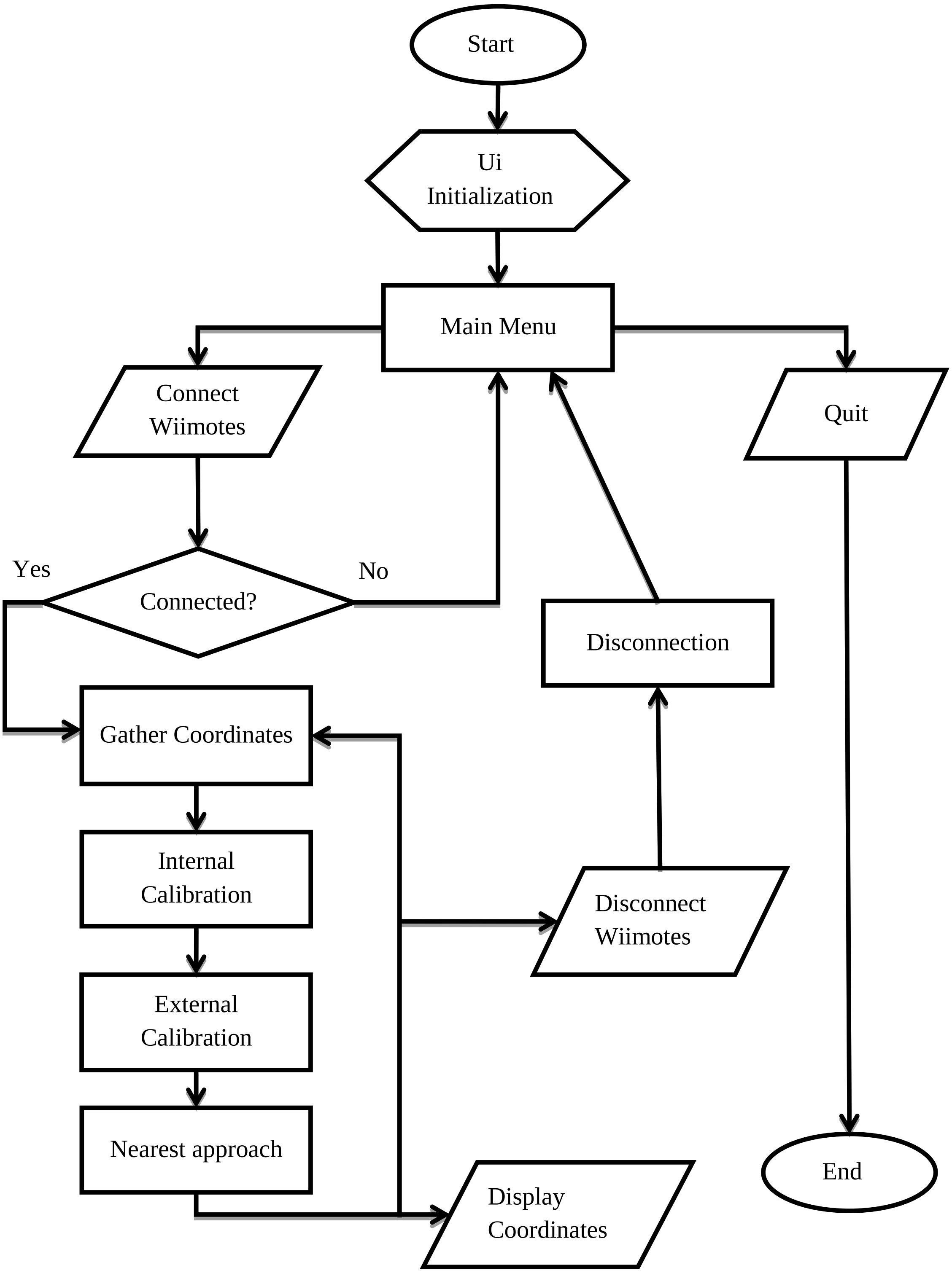
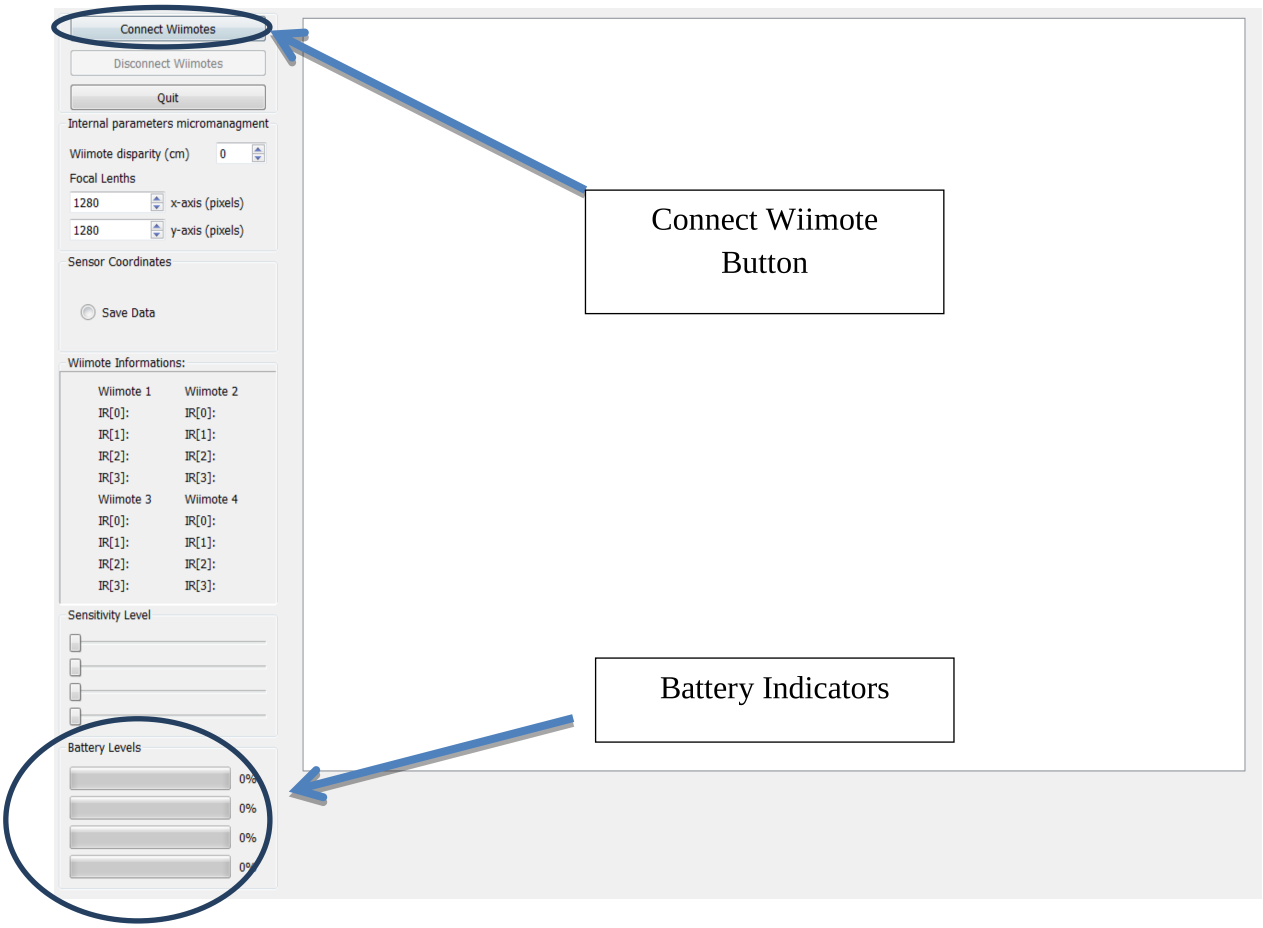


Fig. 3.18 A flowchart of the WiiIRLocator application software.

According to the flowchart given in Fig. 3.18, the recognition and connection of the Wiimotes happens when the “Connect” button is pressed. When the user starts the application she/he presses the “Connect” button to initiate the connection process that lets the application find the paired Wiimote devices through Bluetooth HID. After a brief period, if the process is successful, the Wiimote LEDs will inform the user which one is the first and which one is the second Wiimote by lighting up accordingly. Furthermore, the battery indicator will show the energy amount remaining in each Wiimote (Fig. 3.19).



Wiimote disparity (cm) 0

Focal Lenths

1280 x-axis (pixels)

1280 y-axis (pixels)

Battery Levels

0%

0%

0%

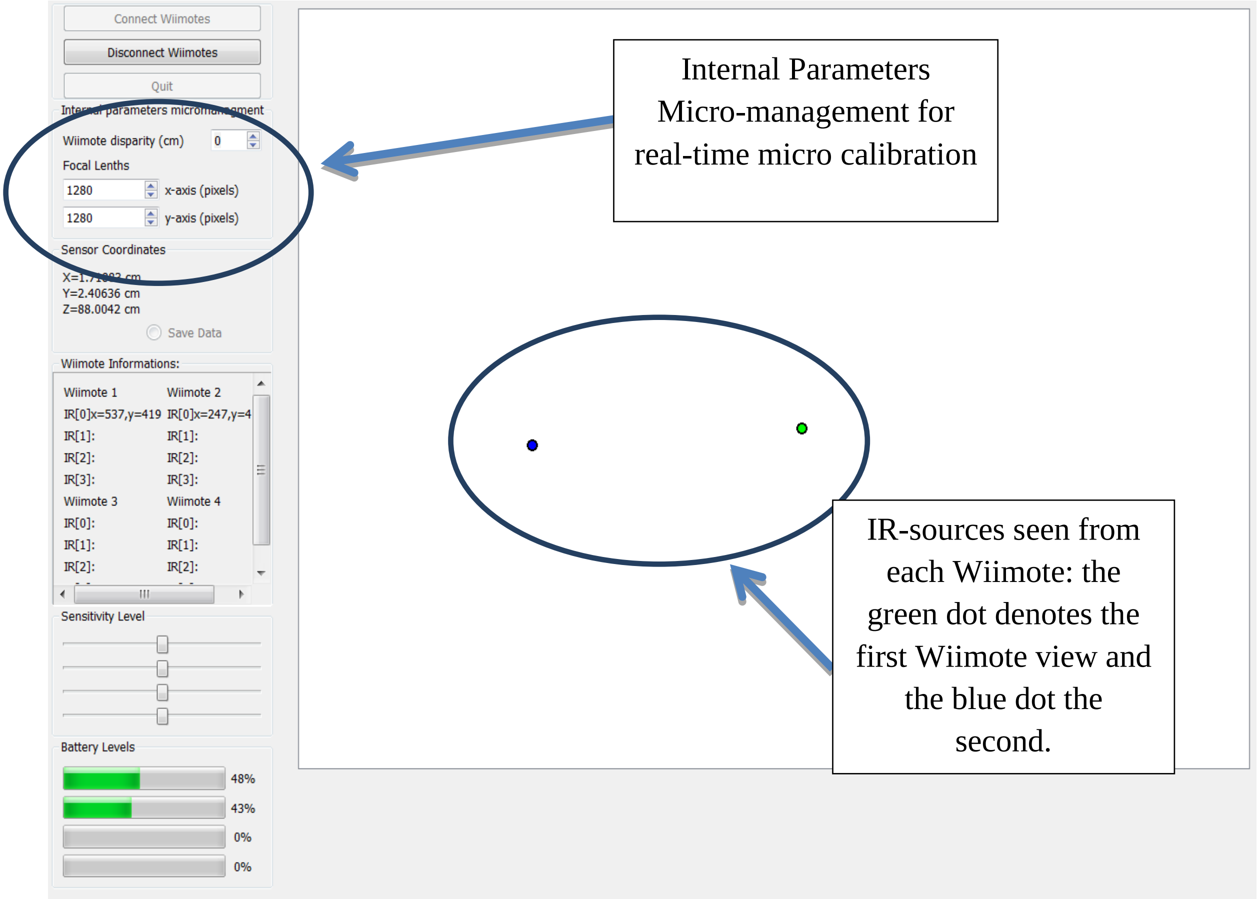
0%

Connect Wiimote Button

Battery Indicators

Fig. 3.19 The GUI of the implemented application software: the “Connect” button initiates the connection with the Wiimotes found during the Bluetooth pairing process. The battery indicator informs the user about the electric energy amount available in each connected Wiimote.

After connection has taken place, the user sets the Wiimotes in parallel and gives the disparity in centimeters to the “Wiimote Disparity Spinner”. By default, when the application starts, the distance between the two Wiimotes is set to be twenty centimeters apart. However, the user can change that according to the needs of the specific application under consideration by entering the amount in centimeters in the spinner called “Wiimote Disparity”. The user has also the ability to change the focal lengths of each camera for micro-calibration according to the application needs in real-time. If left be the first and right be the second Wiimote, the user can start acquiring coordinates by moving the IR emission bulb. The stimulus of the IR irradiation will be shown into the graphical view. Graphical view has an analysis of 1024x768 according to the Wiimote camera oversampled analysis. The green dot on screen denotes the camera pixel which gets most irradiated by the IR emission bulb from the first Wiimote and the blue dot the pixel of the second camera, respectively (Fig. 3.20).



Wiimote disparity (cm) 0

Focal Lenths

1280 x-axis (pixels)

1280 y-axis (pixels)

X=1.7[illegible] cm
Y=2.40636 cm
Z=88.0042 cm

Wiimote 1	Wiimote 2
IR[0]x=537,y=419	IR[0]x=247,y=4
IR[1]:	IR[1]:
IR[2]:	IR[2]:
IR[3]:	IR[3]:
Wiimote 3	Wiimote 4
IR[0]:	IR[0]:
IR[1]:	IR[1]:
IR[2]:	IR[2]:

48%
43%
0%
0%

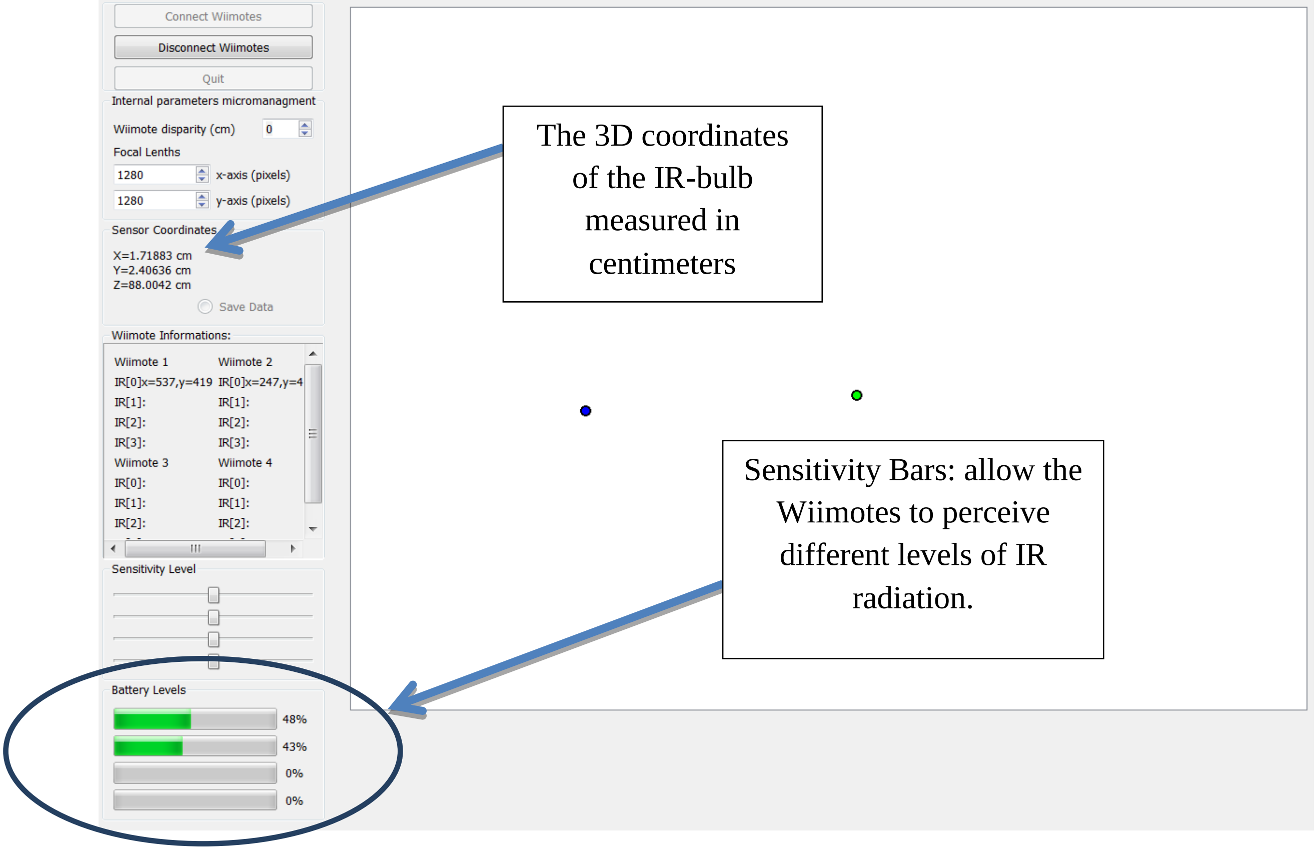
Internal Parameters Micro-management for real-time micro calibration

IR-sources seen from each Wiimote: the green dot denotes the first Wiimote view and the blue dot the second.

Fig. 3.20 When the IR-LED array is detected, the application will create a graphical representation of the source by creating a blue and green dot, which indicates the pixels that get stimulated from each Wiimote view of the scene.

When the 2D image coordinates are gathered in the form of pixels in the image planes of each Wiimote, the application turns the points into 3D coordinates by adding the focal length of each camera as the third dimension and calculates the coordinates of the IR-bulb array (target).The results are given in centimeters at the “Sensor Coordinates” panel (Fig. 3.21).

Last but not least, infrared sensitivity sliders with five levels (default value is three) were added, which allows more stimulus of infrared light to be recognized from the Wiimotes. This is a built-in feature of the CCD sensor used by the Wiimote and allows the CCD to gather five distinct levels of IR radiation, adapting in different lighting conditions. However, sliding the sensitivity above the default level can capture infrared light from sources like lamps or the sun that emits a lower level of infrared light.



Wiimote disparity (cm) 0

Focal Lenths

1280 x-axis (pixels)

1280 y-axis (pixels)

X=1.71883 cm
Y=2.40636 cm
Z=88.0042 cm

```
Wiimote 1        Wiimote 2
IR[0]x=537,y=419 IR[0]x=247,y=4
IR[1]:           IR[1]:
IR[2]:           IR[2]:
IR[3]:           IR[3]:
Wiimote 3        Wiimote 4
IR[0]:           IR[0]:
IR[1]:           IR[1]:
IR[2]:           IR[2]:
```

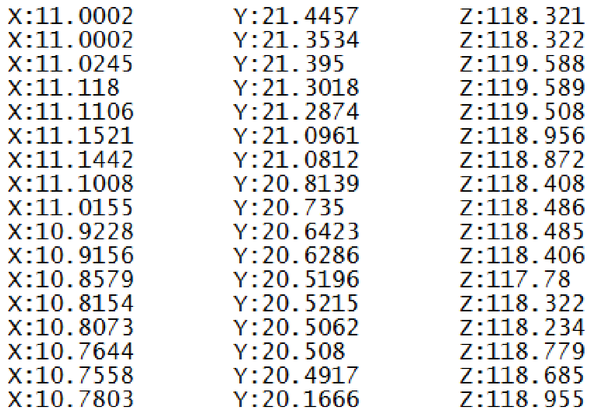
48%
43%
0%
0%

The 3D coordinates of the IR-bulb measured in centimeters

Sensitivity Bars: allow the Wiimotes to perceive different levels of IR radiation.

Fig. 3.21 The result of the coordinates of the IR-array (target) is given in centimeters at the "Sensor Coordinates" panel. Also, the user has the ability to choose from five different levels of IR CCD sensitivity.

Finally, the 'Save Data' button can be marked in order to store the coordinates of the current use and transmitted for input to the actual VR headset (Fig. 3.22). When the 'Save Data' button is marked, the application will create a text file in its folder, called "Data.txt".



```
X:11.0002      Y:21.4457      Z:118.321
X:11.0002      Y:21.3534      Z:118.322
X:11.0245      Y:21.395       Z:119.588
X:11.118       Y:21.3018      Z:119.589
X:11.1106      Y:21.2874      Z:119.508
X:11.1521      Y:21.0961      Z:118.956
X:11.1442      Y:21.0812      Z:118.872
X:11.1008      Y:20.8139      Z:118.408
X:11.0155      Y:20.735       Z:118.486
X:10.9228      Y:20.6423      Z:118.485
X:10.9156      Y:20.6286      Z:118.406
X:10.8579      Y:20.5196      Z:117.78
X:10.8154      Y:20.5215      Z:118.322
X:10.8073      Y:20.5062      Z:118.234
X:10.7644      Y:20.508       Z:118.779
X:10.7558      Y:20.4917      Z:118.685
X:10.7803      Y:20.1666      Z:118.955
```

Fig. 3.22 The measured 3D coordinates of a moving object, gathered into the Data.txt file for use with the HMD software.

4. Experimental Verification

4.1 Experimental Set Up

Experimental tests were conducted, in order to determine the stability and accuracy of the proposed system. The experimental setup was analogous to that described in Chapter 3. The two Wiimotes were set on top of a 1.7 m high bookshelf with 20 cm distance between them (Fig. 4.1). It is expected that normal users generally will not jump or crouch more than fifty centimeters. Precision and accuracy of the system depends on the experimental set up, in terms of the position and orientation of the IR cameras employed. The results presented next have been obtained with the hardware described in Chapter 3 (Fig. 4.2).Two different tests have been carried out to characterize the tracking performance of the proposed system one by measuring sets of real-time coordinates (Fig. 4.3) and one by connecting the Oculus Rift to Unity so that it can operate in cooperation with the proposed positioning system for evaluation of static jitter. Oculus Rift is an HMD device developed by Oculus VR (Fig. 4.4). We used Unity graphics engine to provide the user with a simple 3D virtual room, recreated with the exact dimensions of our workspace for simplicity reasons (Fig. 4.5).

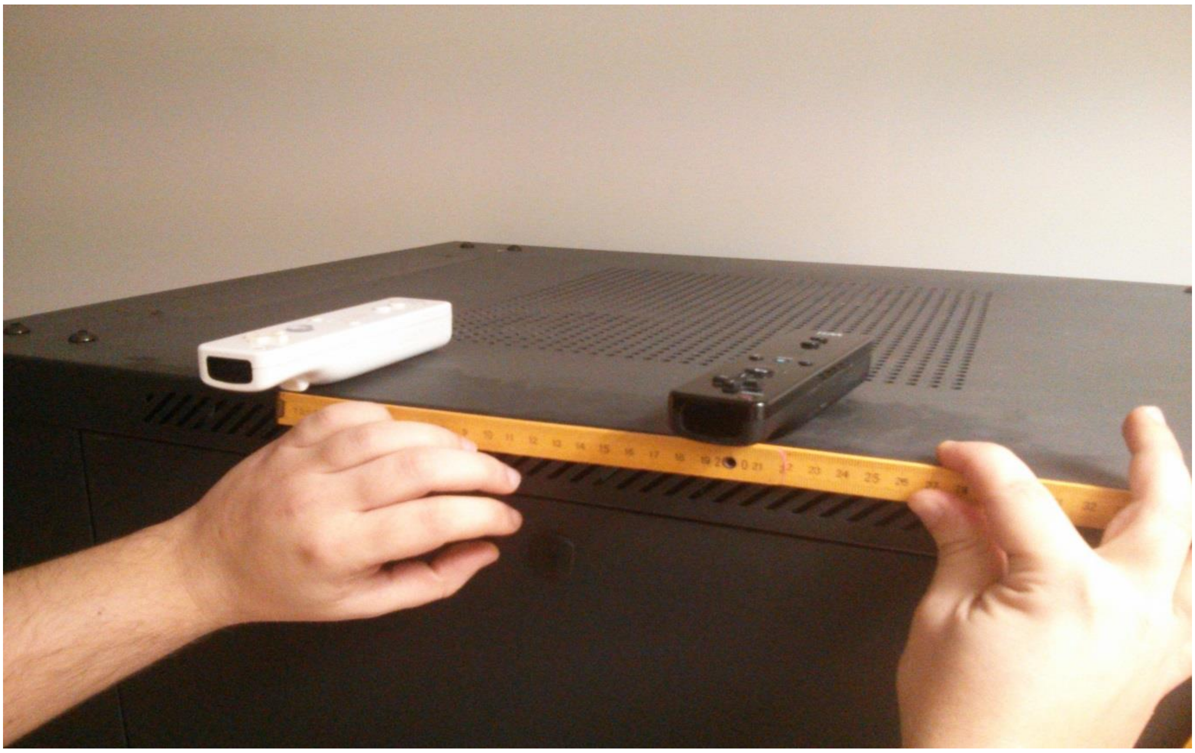


Fig. 4.1 Measuring the distance between the two Wiimotes before the data collection.

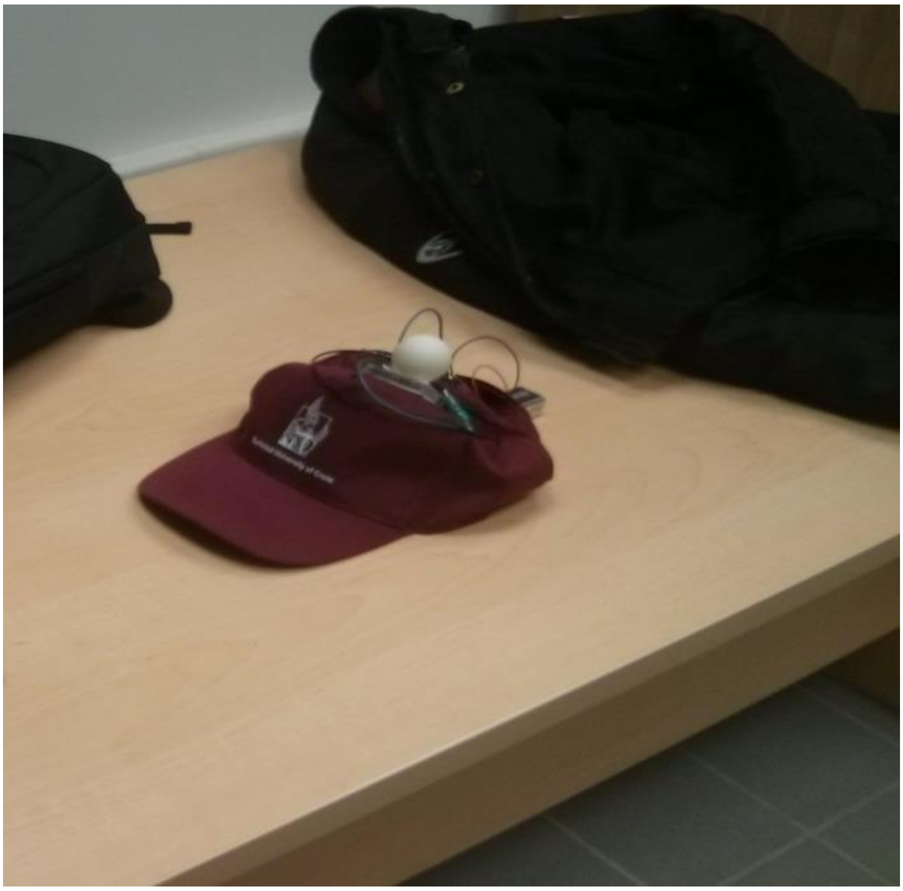


Fig. 4.2 The hat where the IR-LED array bulb was installed on.

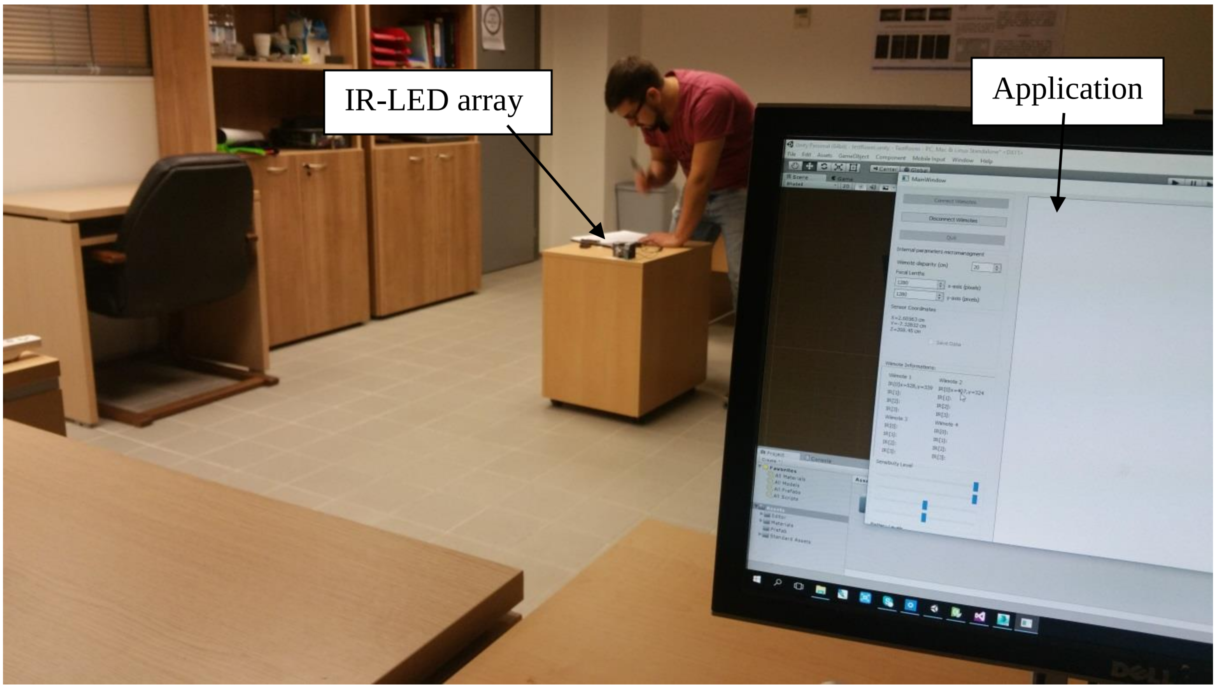


Fig. 4.3 Gathering data in the experiment environment: the IR-LED array on top of a roller table (middle) and the software developed (right).

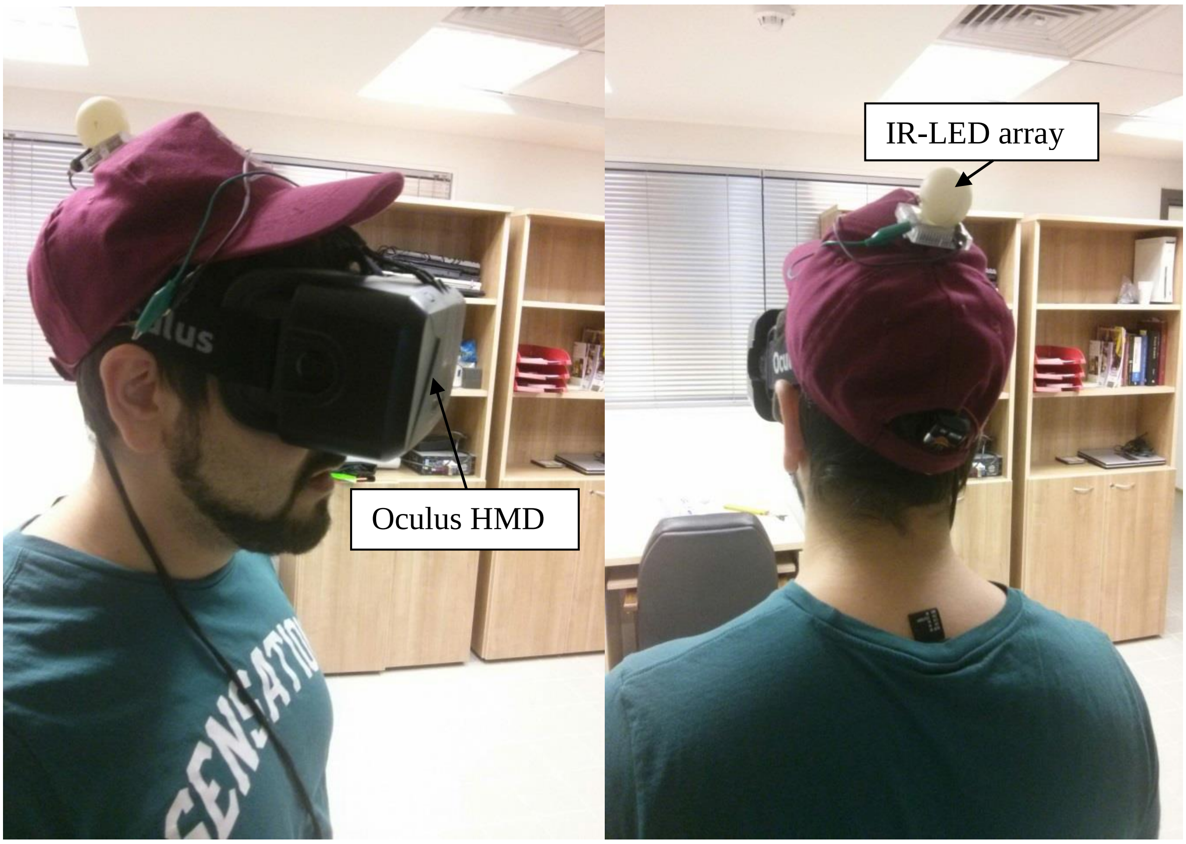


Fig. 4.4 A user of the proposed system wearing a hat with the IR-LED array bulb installed on top and the Oculus Rift HMD.

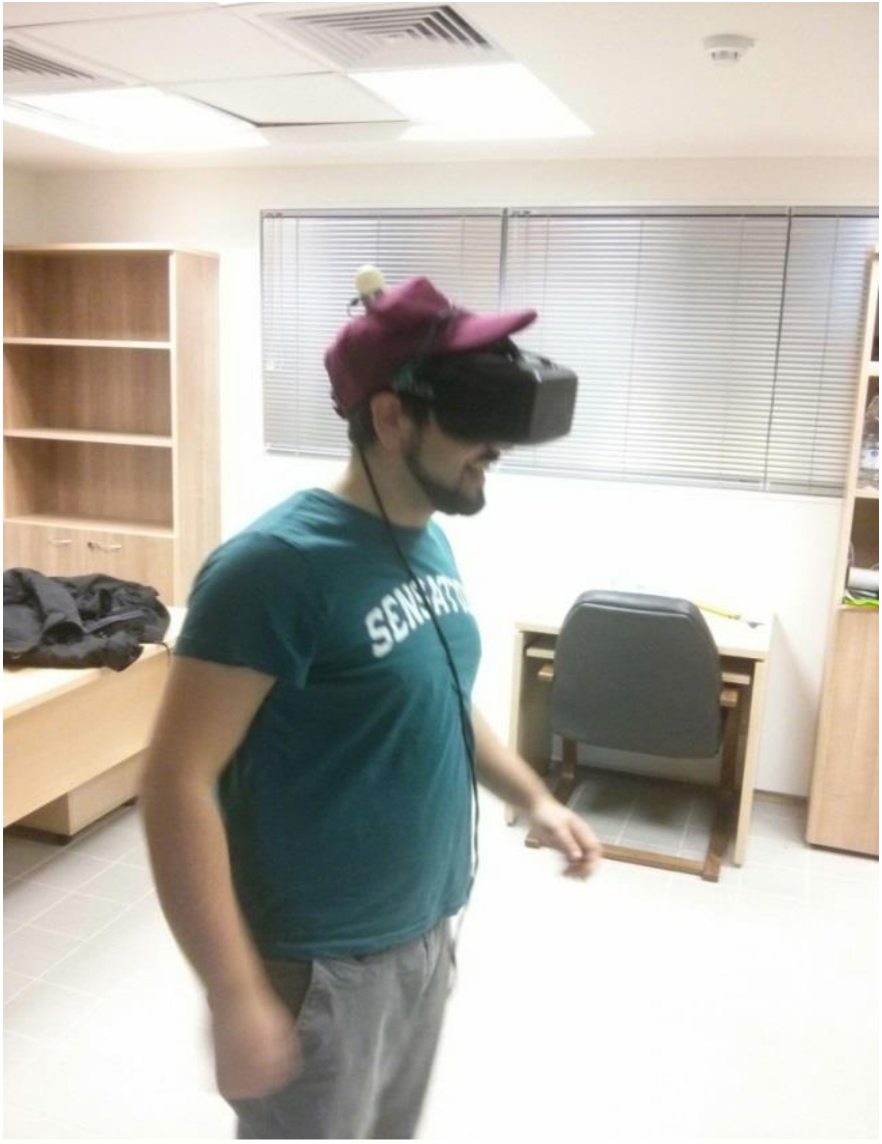


Fig. 4.5 Testing the proposed system with Oculus HMD.

4.2 Error Analysis

The performance of the proposed positioning system has been evaluated by measuring the 3D spatial coordinates of a single IR-LED marker in various different positions within the working volume.

First we separated the depth of the working volume in three levels by placing adhesive tapes on the floor and marking the distances. Level one is one meter distance from our system coordinates origin, while level two, two meters and level three, three meters, respectively (Fig. 4.6). It should be noted that we could get coordinates up to 5 meters but after 3.5 meters our system was beginning to lose focus of our IR-LED single marker. Next we placed our marker on top of a roller table (Fig. 4.3). For each depth level (z-axis), we divided the x-axis (horizontal axis of Wiimotes) into several positions. The diversity of the x-axis range is a result of the working volume which depends on the camera's view scope (see paragraph 3.6).

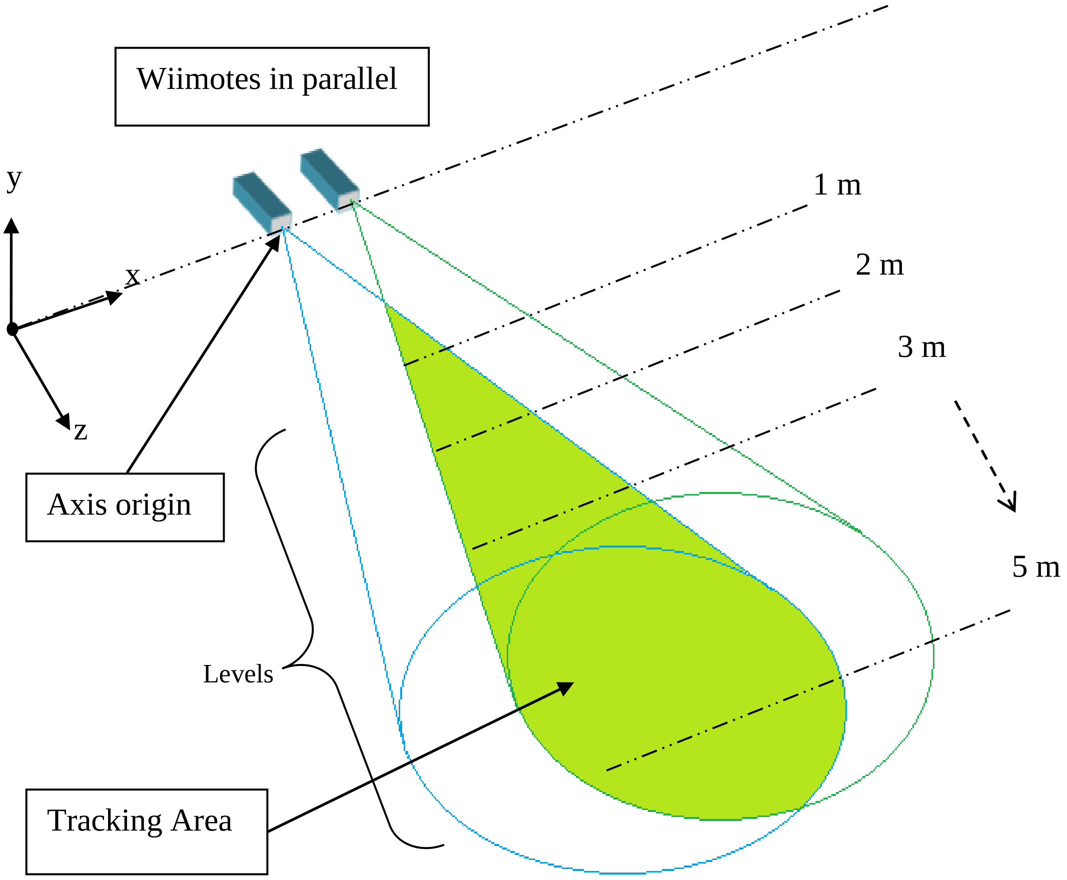


Fig. 4.6 The green area denotes the actual tracking volume of the proposed system. For each depth level on the z-axis, we divided the x-axis into several positions.

By solving the equations (3.29) and (3.32) introduced in paragraph 3.6, given that the distance of the two Wiimotes in parallel relative to the x-axis was set equal to 20 cm, it results that in level one (i.e. for 1 m distance in the z-axis) the measuring range of the x-axis will be 62.8 cm and for the y-axis 63 cm. In level two (i.e. for 2 m distance in the z-axis), the measuring range of the x-axis will be 145.6 cm and for the y-axis 126.1 cm. In level three (i.e. for 3 m distance in the z-axis), the measuring range of the x-axis will be 228.5 cm and for the y-axis 189.1 cm.

In order to evaluate the performance of the proposed system, the absolute measurement error and the mean absolute error were calculated using the following equations:

$$Absolute\ error\ (\%) = \frac{|V_{real} - V_{measured}|}{range} * 100\% \qquad (4.1)$$

$$Mean\ absolute\ error\ (\%) = \frac{(\frac{\sum n |V_{real} - V_{measured}|}{n})}{range} * 100\% \qquad (4.2)$$

where:

V_{real} – is the real value of the distance measured in centimeters
$V_{measured}$ – is the distance measured by the proposed system
$range$ – is the total measurement in the corresponding axis, as analyzed above
n – is the total number of measurements.

Fig. 4.7 and Fig. 4.8 display the measurement errors of seven different positions in the x-axis at one meter depth in the z-axis. The measurement points of Fig. 4.7 have a zero distance in the y-axis, while in Fig. 4.8 the measurement points have a 22 cm distance on the y-axis. The mean absolute error for the results presented in Fig. 4.7 was calculated to be equal to 0.4887%, which corresponds to a 0.3 cm error, while in Fig. 4.8 the mean absolute error percentage was calculated to be equal to 0.3906%, corresponding to a 0.24 cm error.

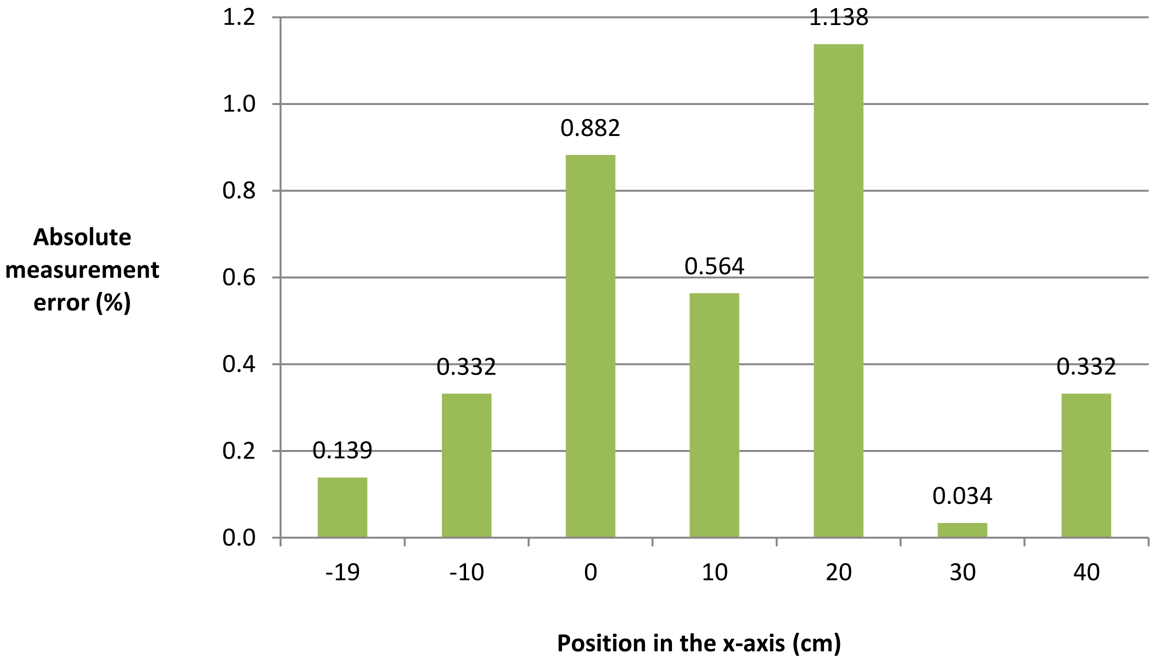


Fig. 4.7 The resulting measurement error in case of 1 m depth, a 62.8 cm measurement range in the x-axis and zero distance in y-axis.

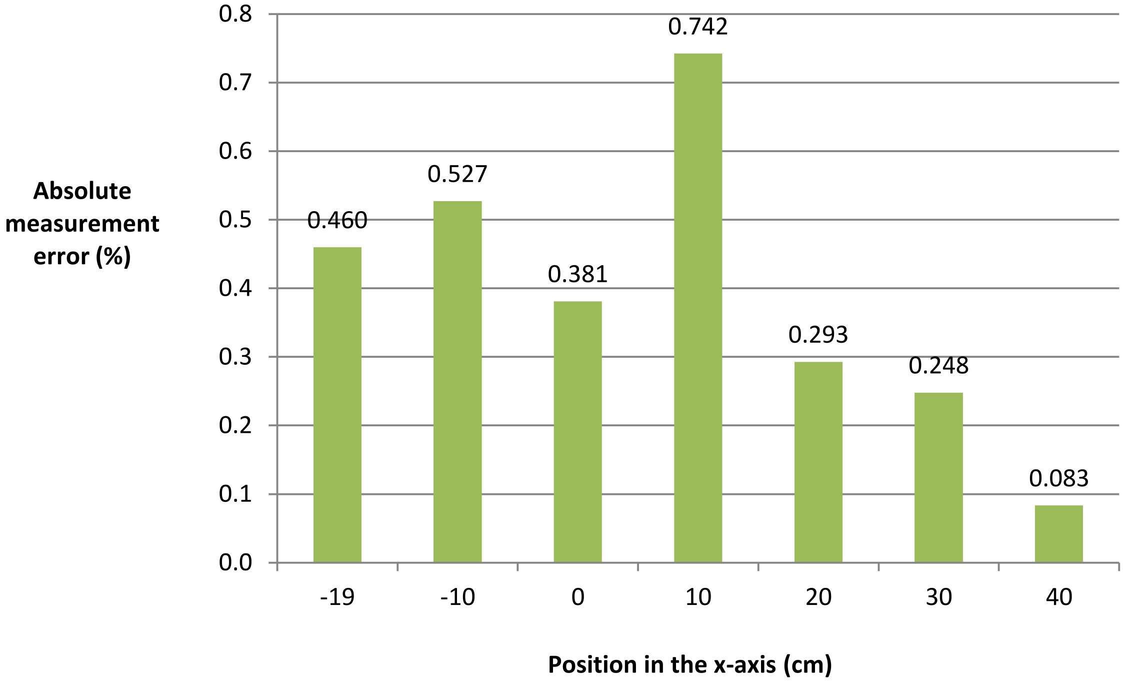


Fig. 4.8 The resulting measurement error in case of 1 m depth, a 62.8 cm measurement range in the x-axis and 22 cm distance on the y-axis.

Fig. 4.9 displays the absolute measurement errors of various different positions in the x-axis at two meters depth in the z-axis and with zero distance in the y-axis. The mean absolute error percentage measured was 0.5491% which corresponds to a 0.79 cm error. The measurement errors of various different positions in the x-axis at three meters depth in the z-axis and with zero distance in the y-axis are plotted in Fig. 4.10. In this case, the mean absolute error was calculated to be equal to 0.4874%, corresponding to a 1.11 cm error.

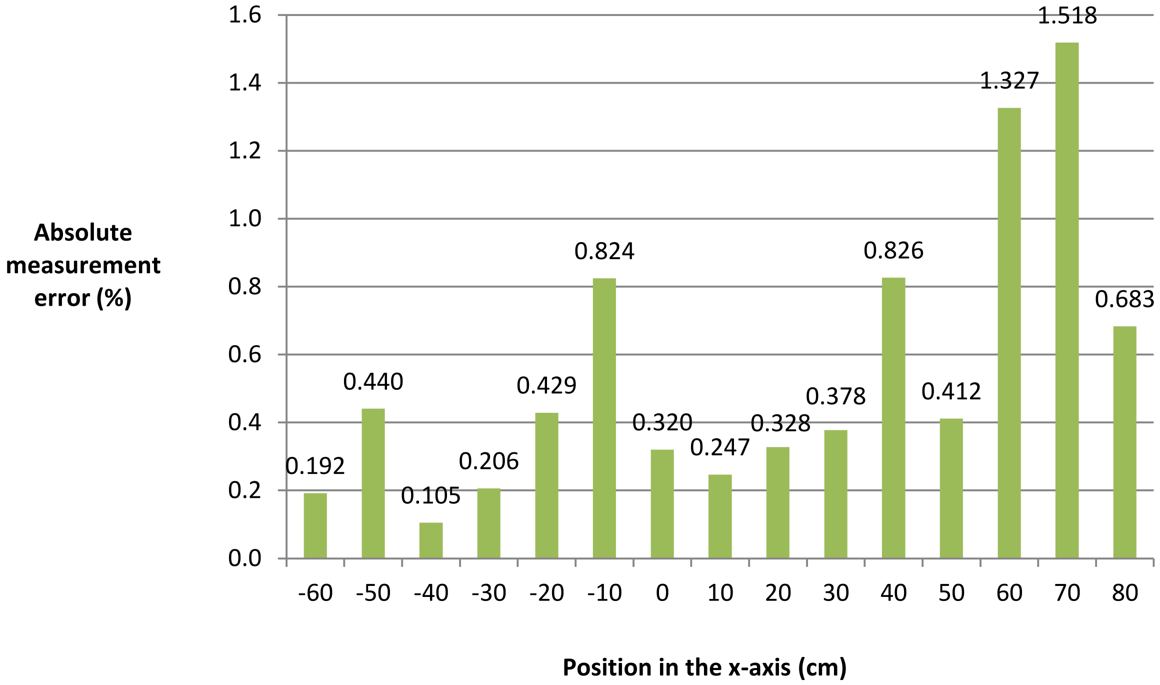


Fig. 4.9 The resulting measurement error in case of 2 m depth and a 145.6 cm measurement range in the x-axis.

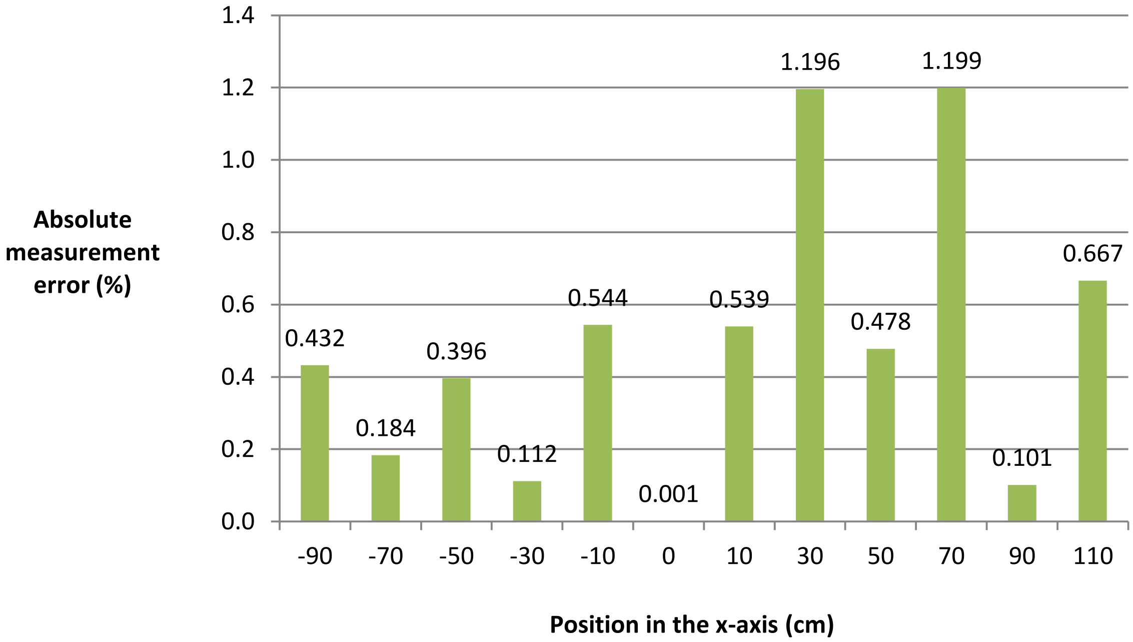


Fig. 4.10 The resulting measurement error in case of 3 m depth and a 228.5 cm measurement range in the x-axis.

The absolute errors of measurements performed exclusively along the z-axis (i.e. depth), are presented in Fig. 4.11. The distances on the x- and the y-axis, respectively, have been set equal to zero. We measured six different positions in a total range of three meters. The mean absolute error percentage was calculated to be equal to 0.4568%, which corresponds to a 1.14 cm error.

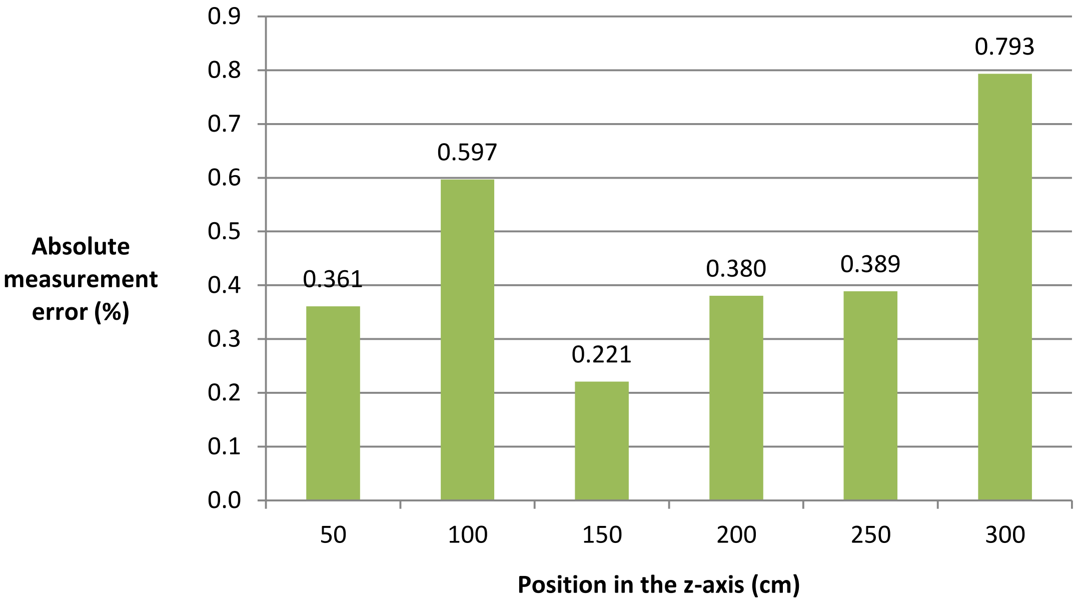


Fig. 4.11 The resulting measurement error along the z-axis.

It concludes from the results of the mean absolute error in each level that the accuracy of the proposed measurement system decreases the further away the infrared light source is from the corresponding optical sensors.

4.3 Latency

Continuous and smooth tracking is essential to applications involving human motion. Additionally, in virtual reality systems, it is of great importance to provide continuity of the user movement over time for realism reasons.

As for optical tracking systems, continuity is related to camera occlusions and can be easily solved by accurate camera placement. In detail, when the line of sight between the LED marker and the cameras is lost, the marker cannot be successfully tracked. On the other hand, smooth tracking is associated with the sampling frequency of the data-acquisition system. This is one of the main reasons why consumer electronic devices cannot be used freely for optical tracking, due to their limitation of low sampling rates.

For measuring the delay between data acquisitions, we used the “QElapsedTimer” build–in library of the Qt framework. The experimental tests were performed by using a PC operating under Windows 7 and equipped with a 3.3 GHz AMD Athlon X3 455 CPU, 4 GB RAM and an HD6770 graphics card. The necessary delay/latency for position reconstruction, including the write-to-file access operation of the proposed system, was measured to be equal to 0.336112 ms. This indicates that our system has an update rate of about 3000 Hz. However, each Wiimote enables a maximum sampling rate of 100 Hz, thus limiting the actual refresh rate of the entire measurement system at 100 Hz. This update rate allows for a smooth reconstruction of the IR-bulb trajectory.

5. Concluding Remarks

In this thesis, the development of a very affordable infrared-optical tracking system was presented, which was based on Wiimotes for immersive HMD virtual reality. By using two Wiimotes, we can apply stereo-vision techniques to acquire 3D tracking data from individual IR emitters. According to the experimental results, the proposed system accuracy is acceptable for virtual reality applications where slight inaccuracies will not impede user task performance in the virtual environment. An example of such an application is a virtual reality game. The cost of the whole system is less than 100 €, without including the PC used for tracking and the 3D virtual reconstruction software. As a result, the proposed system is a fine solution for virtual reality (or augmented reality) position tracking regarding entertainment applications. Although there is space for a lot of improvements, the proposed system can be beneficial for small private laboratories and homemade solutions.

Even though the proposed system is enough for its purpose, several improvements and modifications can take place. These improvements target the main flaws that were observed during the implementation and experimental processes.

There is a plentiful of hardware and software modifications that can be utilized. The Wiimote infrared camera has a very narrow viewing angle. This of course has to do with the main purpose of Wiimote as a handheld cursor device for the Wii console. As a result, in case of short distances, the targeted object may not be detected by the CCD sensor. However, the viewing angle can be expanded by using wide-angle lens as mentioned in [32], where the authors used 37mm lens named “pro digital precision super wide 0.45x AF of Rowa Japan” for widening the viewing angle (Fig. 5.1). Since, theoretically, this modification should have an impact on the camera image distortion, a further investigation of this issue should also take place.

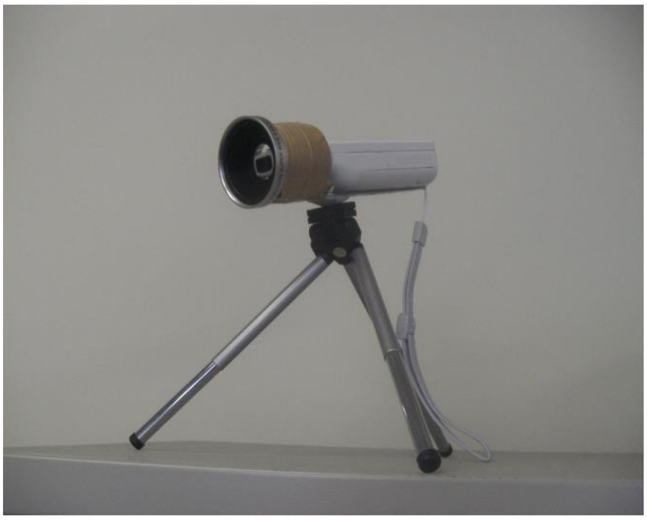


Fig. 5.1 Viewing range expansion by using wide-angle lens [32].

To improve the viewing angle of our IR-array, a white plastic ball was used as mentioned in Chapter 3. However, this idea has a major impact on the brightness of the IR LEDs. The plastic ball used was made of semi-transparent plastic that doesn't disperse the light in an even way. A simple solution would be to configure the IR bulb with more than six IR LEDs, evenly distributed inside the ball. Another solution to disperse light in an omnidirectional way is to use diffusion lens on top of the IR LEDs.

Although two Wiimotes are enough for 3D coordinates acquisition, we can achieve better space coverage by utilizing more than two Wiimotes in pairs, since larger working volumes can be monitored. In this way we extend the measuring range and cope with a wider range of occlusion conditions. Nonetheless, when using multiple cameras, synchronization of data gathering could become a critical factor for flawless positioning. Additionally, by using more Wiimotes we can extend the number of markers (moving targets) tracked simultaneously given that only four IR targets can be monitored by each pair of Wiimotes.

IR illuminators and reflective tags could be added for about 100 to 200 € of additional cost. In this concept, instead of using active markers (i.e. IR-LED bulbs) we make use of passive markers that reflect the light provided by them through IR illuminators. Passive markers consist of reflective materials and more details about their construction can be found in [33].

In lateral stereo vision, two cameras are displaced horizontally in order to acquire two different views of a scene. This scenario of stereo vision is similar to the human binocular vision. When the cameras displacement changes to a non-parallel way (i.e. convergent configuration etc.), the system follows a different geometry, which is called "epipolar geometry". Given two or multiple cameras looking at the same scene from different angles, there are a lot of geometric relations that are associated with the 3D points gathered and their projections. These relations provide certain constraints between the image points and can be used for 3D coordinates acquisition. Thus, when multiple cameras are used in different, distinct locations, occlusion problems can be eliminated.

Kalman filtering, commonly known as linear quadratic estimation, is an excellent way for avoiding inaccuracies from measurements. With this algorithm, instead of using a single measurement to determine the outcome, series of measurements are observed and processed.

The data containing statistical noise or other inaccuracies are analyzed and estimation is produced, which tends to be more accurate. Kalman filtering is a widespread technique applied to optical positioning systems.

6. References

[1] R. Mautz, “Indoor positioning technologies”, ETH, Zurich, 2012.

[2] http://www.ar-tracking.com/products

[3] Robert Repas, “Sensor Sense: Pulse ranging technology”, 2013, http://machinedesign.com/sensors/sensor-sense-pulse-ranging-technology

[4] http://www.apislabs.us/soter.html

[5] https://kn0l.wordpress.com/465-2/

[6] Zoltan Koppanyi, Charles K. Toth and Dorota A. Grejner-Brzezinska, “Positioning in Challenging Environments Using Ultra-Wideband Sensor Networks”, 2015, http://gpsworld.com/innovation-where-are-we/

[7] http://landsurveyorsunited.com/photo/polar-method-surveying

[8] Erik Dahlgren, Hasan Mahmood, “Evaluation of indoor positioning based on Bluetooth Smart technology”, M.Sc. Thesis, Chalmers University of Technology, Goteborg, Sweden, June 2014.

[9] Jiménez, Antonio R., Francisco Zampella, and Fernando Seco. "Improving inertial pedestrian dead-reckoning by detecting unmodified switched-on lamps in buildings" in Sensors vol.14, issue 1, pp. 731-769 Jan. 2014.

[10] Meng Yu, Zhilin Li, Yongqi Chen and Wu Chen “Improving integrity and reliability of map matching techniques Positioning”, in Journal of global positioning systems, vol.5, No 1-2, pp.40-46, 2006.

[11] https://en.wikipedia.org/wiki/Thermography

[12] Karl Sanford, “Smoothing Kinect Depth Frames in Real-Time”, 2012, http://www.codeproject.com/Articles/317974/KinectDepthSmoothing

[13] https://en.wikipedia.org/wiki/Animal_echolocation

[14] http://www.danbygroup.com/rfid

[15] http://telecom.dei.unipd.it/pages/read/48/

[16] J.C. Lee, “Hacking the Nintendo Wii Remote”, in IEEE Pervasive Computing, vol. 7, no. 3, pp. 39-45, July-Sept. 2008.

[17] http://www.wiibrew.org/

[18] http://keera.co.uk/blog/2014/05/27/interacting-haskell-games-using-wiimote/

[19] http://www.gizbot.com/how-to/what-is-wiimote-how-to-add-wiimote-to-bluetooth-places.html

[20] http://www.mirror.co.uk/news/uk-news/jamie-redknapp-in-smash-it-wii-111132

[21] https://www.sparkfun.com/tutorials/43

[22] http://www.thedoityourselfworld.com/Free-Online-MCD-Millicandela-To-Lumens-Converter.php

[23] http://wiibrew.org/wiki/List_of_Working_Bluetooth_Devices.

[24] R. I. Hartley, A. Zisserman, “Multiple View Geometry in Computer Vision”, Cambridge University Press, ISBN: 0521623049, 2000.

[25] https://en.wikipedia.org/wiki/Pinhole_camera

[26] Z. Zhang. "A flexible new technique for camera calibration" in IEEE Transactions on pattern analysis and machine intelligence vol. 22, issue 11, pp. 1330-1334, Nov. 2000.

[27] http://tutorial.math.lamar.edu/Classes/CalcIII/EqnsOfLines.aspx

[28] Dan Sunday, “Distance between Lines and Segments”, 2010, http://geomalgorithms.com/a07-_distance.html

[29] https://www.mathworks.com/matlabcentral/newsreader/view_thread/246420

[30] https://github.com/rpavlik/wiiuse

[31] http://en-americas-support.nintendo.com/app/answers/detail/a_id/2815/~/how-to-sync-a-wii-remote-with-a-wii-console

[32] Yoshiaki Nakano, Takeo Tatsumi, Kiyoshi Tajitsu, “Wiimote positioning system - an epoch-making system of indoor position detection”, in IFIP EduTech2009, pp. 1-9, 2009.

[33] S.Amici, A.Sanna, F.Lamberti, B.Pralio, “A Wii remote-based infrared-optical tracking system” in Entertainment Computing, vol. 1, pp. 119–124, 2010.

[34] Zhihua Ly, Zhiyi Zhang, “Build 3D Laser Scanner Based on Binocular Stereo Vision”, in 2011 International Conference on Intelligent Computation Technology and automation (ICICTA), vol. 1, pp. 600-603, March 2011.

[35] Chow, Y., “3D spatial interaction with the Wii remote for head-mounted display virtual reality” in Proceedings of World Academy of Science, Engineering and Technology: Vol.38, pp. 381-387, 2009.

[36] Yildiz, Alparslan, Abdullah Akay, and Yusuf Sinan Akgul. "Wii remote calibration using the sensor bar", in CVPR 2011 workshops IEEE, pp. 7-12, June 2011.