POLITECNICO DI TORINO

Corso di Laurea Magistrale
in Ingegneria Meccatronica

Tesi di Laurea Magistrale

Vision and Inertial Data Fusion for Collaborative Robotics

POLITECNICO
DI TORINO

Dipartimento
di Automatica e Informatica

Relatori
prof. Marcello Chiaberge
prof. Sarah Cosentino

Candidato
Anna Grosso [s253169]

Anno accademico 2019-2020

Abstract

Robots capable of engaging in collaborative behaviours with humans, widely known as cobots, are characterized by incredibly complex requirements and are one of today's major challenges in the robotics field. In order to meet the rather strict accuracy requirements needed to ensure human safety and to gather context information useful for intelligent human-robot collaboration, these robots must adequately localize human operators who move freely in the robotic workplaces. In today's industrial environments, this objective can be achieved by adopting sophisticated sensory devices like lasers, ultrasounds or vision systems. However, human tracking can be particularly difficult in presence of occluding factors that could severely affect vision-based or light-based approaches and in unconstrained conditions like crowded spaces.

This thesis analyzes the integration of inertial measurement units and a vision system in order to improve the human localization for collaborative robotics purposes. More in detail, this work first shows how the human upper body can be independently reconstructed by means of an inertial motion capture system and of a stereoscopic vision system. In order to take advantage of both types of sensors, the measurements of these systems are then combined using a two-step Kalman filter fusion algorithm.

The approach is first validated by simple calibration movements. Then, some complex movements are considered in order to verify the effectiveness of the framework. In particular, two different categories of movements are experimentally tested: i) short movements where the subject comes back to a rest condition every few seconds and ii) long movements where the subject performs a long motion task without going back to the rest position until the end. Experimental results show that the presence of IMU sensors in addition to cameras can compensate for the typical drift of IMU sensors and effectively improve the spatial perception of the robot. This result could be of great interest not only for direct interaction tasks between humans and robots, but also in the characterization of advanced robotic cells, where human behaviour can be gradually learned and the use of IMU sensors can be finally disregarded, in favour of a pure three-dimensional reconstruction through artificial vision.

Acknowledgements

First, I would like to thank my thesis supervisors, prof. Marcello Chiaberge and prof. Sarah Cosentino, for their guidance and their support throughout the development of this thesis.

My sincere gratitude to prof. Atsuo Takanishi and prof. Sarah Cosentino for the incredible opportunity to intern at a renowned institution such as Waseda University and to all the members of the WB and the Musical teams of prof. Takanishi Laboratory, who welcomed me and made me feel at home even on the other side of the world and without whose assistance the experimental part of this thesis would not have been completed.

An especially heartfelt thank you to my parents, who have always supported me in any possible way, believing in me and encouraging me in every situation, to my brother who constantly inspires me to aim higher and higher and to my whole family for the love they show me everyday.

I would like to thank also my second family: all the people who lived on the fifth floor of Collegio Einaudi - Sezione San Paolo in the last six years, for their endless support and inspiration, for creating such a vibrant and loving community and for the countless memories that I will cherish forever, and all my closest friends for their ability to make me smile in times of discouragement and for always being by my side whenever I needed them.

Last but not least, I am deeply grateful to Politecnico di Torino, Collegio Einaudi and McKinsey & Company for their financial support that made this Master's thesis project abroad possible.

Contents

List of Tables 6

List of Figures 7

1 Introduction 11
- 1.1 Context . . . 11
- 1.2 Motivation . . . 11
- 1.3 Goals . . . 14
- 1.4 Thesis outline . . . 15

2 Inertial Measurement Units (IMUs) 17
- 2.1 Human Skeleton Reconstruction from IMU Sensors: Problem Formalization 17
  - 2.1.1 Alignment Matrices . . . 18
  - 2.1.2 Initial Rotation Matrices . . . 22
    - 2.1.2.1 The Euler angles . . . 22
    - 2.1.2.2 The quaternions . . . 24
  - 2.1.3 Instantaneous Rotation Matrices . . . 25
  - 2.1.4 Upper Body Trajectory Reconstruction . . . 25
    - 2.1.4.1 Anatomical Constraints . . . 27

3 Cameras 29
- 3.1 Camera model . . . 29
- 3.2 Camera calibration . . . 31
  - 3.2.1 Intrinsic parameters . . . 31
  - 3.2.2 Extrinsic parameters . . . 33
  - 3.2.3 Estimating M . . . 33
- 3.3 Stereo vision . . . 34
  - 3.3.1 The correspondence problem . . . 35
    - 3.3.1.1 The epipolar constraint . . . 35
    - 3.3.1.2 Other constraints . . . 36
    - 3.3.1.3 OpenPose . . . 37
  - 3.3.2 The reconstruction or triangulation problem . . . 39
- 3.4 Rototranslation from the camera world reference system to the global reference system . . . 41

4 The Kalman filter 45

4.1 The discrete Kalman filter . . . 46

4.2 The Kalman filter fusion algorithm . . . 48

5 Experiments 51

5.1 Experiments Instrumentation . . . 51

5.1.1 Board . . . 51

5.1.2 IMU sensors . . . 52

5.1.2.1 Inertial sensors calibration . . . 53

Accelerometer calibration . . . 53

Gyroscope calibration . . . 54

5.1.3 Cameras . . . 55

5.2 Experiments Setup . . . 55

6 Results 59

6.1 Calibration movements . . . 59

6.2 Complex movements . . . 59

6.2.1 IMUs-only reconstruction . . . 62

6.2.1.1 Short movement . . . 62

6.2.1.2 Long movement . . . 63

6.2.2 Vision-only reconstruction . . . 65

6.2.2.1 Short movement . . . 65

6.2.2.2 Long movement . . . 66

6.2.3 IMUs + Cameras reconstruction . . . 66

6.2.3.1 Short movement . . . 66

6.2.3.2 Long movement . . . 67

6.2.4 Results Comparisons . . . 69

7 Discussion 73

8 Conclusions and Possible Applications 75

List of Tables

5.1 Specifications of WB-4R IMU sensors. 52

5.2 Specifications of Logitech HD Pro Webcams C920. 57

5.3 Anthropometric information of experimental subjects. 58

6.1 Mean ICP errors of short and long experiments obtained after the ICP minimization procedure for the three reconstruction methods (IMUs only, vision only, Kalman fusion of IMUs and vision measurements). 69

List of Figures

2.1 Relationship between the three reference systems $\boldsymbol{F}^g$, $\boldsymbol{F}^b$ and $\boldsymbol{F}^i$ for each body segment. The two continuous arrows indicate the two direct transforms between the body and the IMU reference frames (called alignment matrices in the following sections) and between the IMU and the global reference frames (called initial and instantaneous rotation matrices in the following sections). The dashed arrow indicates the indirect transform between the body and the global reference frames, which can be obtained by combining the two direct transforms. 18

2.2 Global reference system (in green), eight body reference systems (in black) and eight IMU reference systems (in red) positioned on the human body. The eight IMU sensors are numbered in red. 19

2.3 Sagittal, coronal and transverse planes. 20

2.4 Right hand calibration movements. In the two top plots the output of the gyroscope, so the angular velocity expressed in semi-turns per second, on the three axes (x,y,z) is plotted in red, green and blue, respectively. In the two bottom plots the output of the accelerometer, so the acceleration expressed as a function of gravity g, on the three axes (x,y,z) is plotted in red, green and blue, respectively. 21

2.5 Torso calibration movements. The rotation is around the IMU 1 x-axis (shown in red). 22

3.1 Pinhole camera model and perspective projection. 30

3.2 Pinhole camera coordinate systems. 30

3.3 Different coordinate systems and intrinsic and extrinsic parameters. 32

3.4 Stereoscopic system formed by two cameras. 35

3.5 The epipolar geometry. 36

3.6 The overall OpenPose pipeline. 38

3.7 The two possible pose output formats supported by OpenPose. 39

3.8 Two camera frames of two different subjects during the experiment sessions and the corresponding two outputs of OpenPose, with the identified joints and limbs. 40

3.9 Side view and top view of the two reference frames. In green, the global reference frame to which the positions determined by the IMU sensors are reported. In red, the camera world reference frame in which the positions determined by the two cameras are found. 42

4.1 The recursive process of the Kalman filter. 45

4.2 The discrete Kalman filter algorithm (from [42]). 48
4.3 The fusion algorithm diagram. 49
5.1 The grid used in the experiments. 51
5.2 WB-4R IMU sensor and IMU placement on the body. 52
5.3 Equipment necessary for the accelerometer calibration. 53
5.4 Correct positioning of theIMU sensor inside the calibration box and horizontal inclination checking using the level gauge. 54
5.5 IMU accelerometer calibration. 55
5.6 IMU gyroscope calibration. 56
5.7 Logitech HD Pro Webcams C920. 56
5.8 Room setup for the experiments. 57
5.9 Experiments setup. 58
6.1 Three-dimensional reconstruction of the first calibration movement. The right arm is rotated forward, along the sagittal plane. Its position in the global reference frame is plotted once every 20 time steps. 60
6.2 Test subject with the arm raised forward at 90 degrees and corresponding reconstruction. 60
6.3 Three-dimensional reconstruction of the second calibration movement. The right arm is rotated sideways, along the coronal plane. Its position in the global reference frame is plotted once every 20 time steps. 61
6.4 Test subject with the arm raised sideways at 90 degrees and corresponding reconstruction. 61
6.5 Three-dimensional reconstruction of the test subject touching the 4 corners of the grid and difference between the 4 corners real and estimated positions in the global reference frame, using inertial measurements only. 63
6.6 Difference between the 25 points real positions (in blue) and their estimated positions using inertial sensors only along the x, y and z directions (in red). 64
6.7 The error evolution in time using inertial sensors only. On the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis the norm of the error vector is plotted. 64
6.8 Three-dimensional reconstruction of the test subject touching the 4 corners of the grid and difference between the 4 corners real and estimated positions in the global reference frame, using cameras measurements only. 65
6.9 Difference between the 25 points real positions (in blue) and their estimated positions using visual data only along the x, y and z directions (in red). . . 66
6.10 The error evolution in time using visual data only. On the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis the norm of the error vector is plotted. 67
6.11 Three-dimensional reconstruction of the test subject touching the 4 corners of the grid and difference between the 4 corners real and estimated positions in the global reference frame, obtained using Kalman fusion of inertial and cameras measurements. 68
6.12 Difference between the 25 points real positions (in blue) and their estimated positions using Kalman fusion along the x, y and z directions (in red). . . . 68

6.13 The error evolution in time using Kalman fusion. On the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis the norm of the error vector is plotted. 69

6.14 Comparison between the error accumulation in the long experiment using IMU sensors only (in red), vision only (in blue) and Kalman fusion (in green). 70

6.15 The camera frame taken at the time instant at which the test subject is touching the tip of the stick number 4 of the board. 70

6.16 The corresponding OpenPose output, in which the joints and limbs are identified. 71

6.17 The corresponding three-dimensional reconstruction obtained using the Kalman fusion algorithm of inertial and vision data. 71

8.1 Safety zones for human-robot collaboration with speed and separation monitoring technology (Courtesy of ABB Robotics). The green one is the safe zone, the yellow one is the warning zone and the red one is the danger zone. 77

Chapter 1

Introduction

1.1 Context

In the recent years, collaborative robotics has started to emerge as one of the main applications in the field of robotics. Collaborative robots (or cobots) were invented by J. Edward Colgate and Michael Peshkin, professors at Northwestern University, in 1996. A 1999 US patent [1] describes a cobot as "an apparatus and method for direct physical interaction between a person and a general purpose manipulator controlled by a computer". In fact, collaborative robots are intended to work alongside humans and to directly engage with them in a shared space. They can be used for social purposes, mainly to facilitate relationships, entertain people and connect them with the outside world, or in industrial environments, to assist human operators and improve their working conditions and the overall efficiency of an assembly line. In contrast to pure industrial robotics, performance requirements in collaborative robotics are looser, since the involved velocities and accelerations in cobots are lower with respect to industrial robots. Nevertheless, collaborative robots are characterized by incredibly complex requirements and are one of today's major challenges in the robotics field.

1.2 Motivation

Even though recently the research in the robotics field has come a long way, researchers are still far from reaching a full collaboration between humans and robots. The main challenges consist in respecting the strict accuracy requirements needed to ensure human safety and to gather context information useful for intelligent human-robot collaboration. To this aim, an essential step is the real-time localization of the human operators who move freely in the robotic workplaces. This entails the spatial perception of the human body, in order to inform the robot about the operators' position at every time instant and to ensure their safety throughout the entire human-robot collaboration.

Human localization is a very challenging task, as human behaviours are commonly affected by a great number of external factors and are often unpredictable. Researchers have been using different methods to deal with this problem and many systems have been developed over the years, which employ different kinds of sensors. In today's industrial environments, researchers have adopted sophisticated sensory devices like lasers, ultrasounds or vision systems. Okada et al. [2] presented a method for recognizing the motion of people using static and mobile laser scanners in an indoor environment. Holban et al. [3] presented an approach on the reconstruction of 3D objects and calculation of their volumes from their 2D ultrasound images, showing how this technique has given excellent results regarding the precise knowledge of the human body. Some researchers reconstructed the human skeleton analyzing the data coming from monocular video sequences: Remondino and Roditakis [4] fitted a pre-defined human model to the recovered 3D data, Loy et al. [5] incorporated limb length and symmetry constraints to obtain a three-dimensional reconstruction of human actions in long image sequences, Chen and Chai [6] constructed a human motion model from a vast collection of preprocessed human motion examples to constrain the solution space and learnt a skeleton model from prerecorded data to minimize the ambiguity of the human skeleton reconstruction, Guler and Kokkinos [7] introduced HoloPose, a method for reconstructing the three-dimensional human body in the wild using a monocular camera, aligning the model-based joint positions 3D estimates and DensePose with their image-based equivalents provided by CNNs and achieving both global consistency and high spatial accuracy of the joint and of the 3D surface estimates. Other researchers used stereo video streams, like Liu et al. [8] who tracked positions of the joints over all the subsequent frames and matched the corresponding joints to the ones tracked on the image sequences from the other camera, reconstructing the skeleton models in a three-dimensional space through triangulation. Depth imaging technology has progressed significantly in the last few years and eventually reached a consumer price point with the launch of Kinect, a cheap RGB-D binocular sensor providing synchronized color and depth images. A comprehensive review of the latest computer vision algorithms and applications based on Kinect can be found in [9]. Since its release, also Kinect has been used for human tracking: Shotton et al. [10] used single depth images for real-time human pose identification, using an object recognition method and developing an intermediate body parts representation that maps the complex pose estimation problem into a simpler per-pixel classification task, while Alexiadis et al. [11] used Kinect depth maps to track dancers skeletons in order to evaluate their movements in real-time in online interaction environments.

Despite all these efforts, human tracking has proven particularly difficult in presence of occluding factors that severely affect vision-based or light-based approaches and in unconstrained conditions like crowded spaces.

Another type of sensors used to address the problem of the human localization are Inertial Measurement Units (IMU). These sensors are widely used as a wearable tool for human motion tracking, motion capture and motion evaluation. A survey on IMU-based human tracking can be found in [12]. For example, Roetenberg et al. [13] designed the Xsens MVN motion capture suit, a cost effective system for full-body human motion capture which is based on state-of-the-art miniature inertial sensors, biomechanical models and sensor fusion algorithms that can record all types of movements, including jumping, running and crawling, and can be used outdoors as well as indoors. Zheng et al. [14] designed Pedalvatar, a cheap IMU-based system that can record the users' full-body motion in real-time using a kinematic model rooted at one foot. With respect to systems based on vision, this IMU-based system ensures more flexibility to capture outdoor activities that

are important for several robotic applications. Kong [15] designed a miniaturized, portable lower body motion capture system named WB-4R for elderly people gait telerehabilitation, while Zhang [16] used inertial sensors to reconstruct only one arm: he investigated overhead throwing to reconstruct the trajectory and the rotation velocities of the throwing arm, as well as the torque and the force imposed on the elbow and shoulder. Lin et al. [17] instead proposed a model that evaluates the performance of surgical movements in the laparoscopic training program, using an ultraminiaturized wearable motion capture system (Waseda Bioinstrumentation system WB-3) to analyze the kinematic data describing the movements of a surgeon's arms.
IMU sensors constitute a good alternative for marker-based optical tracking systems, because their workspace is not limited to a camera's field of view and, unlike cameras, they are not affected by occlusions. Even though inertial sensors are small and integrated, they are rarely used in collaborative robotics applications, since they are affected by a consistent drift error and therefore they alone cannot provide the accuracy required in order to satisfy the safety requirements during collaborative tasks. Furthermore, the positioning of the IMU sensors (by means of strips or incorporated in wearable clothes) does not guarantee a rigid connection with the body segments, with significant repercussions on the estimate of the human limbs position.

A more recent approach consists in integrating the measurements coming from different types of sensors, in order to exploit the advantages of each one of them. For example, Jia et al. [18] explored the combination of high quality images coming from a stereo vision system and fast computation of depth information of Kinect to develop a high resolution 3D image reconstruction system, which has a wide variety of applications including 3D body motion detection, hands tracking and finger gestures. Recently, low-cost inertial measurement units and Kinect techniques have proven to offer a feasible and cost-effective solution for trajectory tracking problems, though each of them still has its own limitations. For example, Destelle et al. [19] fused the joint positions found by the Kinect sensor with the more accurate measurements of body segment orientations provided by inertial sensors, in order to implement a low-cost accurate skeleton tracking, and they achieved a very high level of accuracy. Tian et al. [20] instead integrated the same two types of sensors to perform upper limb motion tracking, with the aim of obtaining robust hand position information. Safeea and Neto [21] investigated the use of a laser scanner and IMU sensors for a human-robot interaction application in a dynamic environment with moving humans and obstacles. The data from the laser scanner and from the inertial measurement units positioned on the human body were fused together to find the position of the subject's torso and the configuration of his upper body, in order to determine the distance between the human and the robot on the fly and to avoid collisions between them. Corrales and Candelas [22] designed a hybrid tracking system for human operators using IMU and Ultra Wide Band (UWB) data fusion by a Kalman filter. Their algorithm exploits the advantages of both technologies: global translational precision from the UWB localization system and high data rates from the motion capture system. In this way, their developed hybrid system is able to track the movements of all the limbs of the user and also to precisely position the user in the environment. Xu et al. [23] instead integrated IMU and UWB data using an unbiased finite impulse response (UFIR) filter. Brodie et al. [24] used inertial sensors and GPS to design a prototype system for the biomechanical analysis of ski racing. Liu et al. [25] designed an innovative data fusion method of INS/GPS navigation systems based on adaptive Kalman filtering for autonomous vehicles navigation. Similar approaches have been proposed for the integration of IMUs and monocular or binocular vision systems. A recent survey from Chen et al. [26] provides an overview of the latest investigations in which vision and inertial measurement units are used simultaneously to perform human action recognition more effectively and a summary of the elements required to accomplish the integration of data from depth and inertial sensors. For example, Nutzi et al. [27] performed the fusion of visual and inertial data to improve the pose estimation of an object and to determine the unknown scale parameter in a monocular SLAM framework. Schmid and Hirschmuller [28] designed a system that computes high quality depth images and estimates the ego-motion by fusing key frame-based visual odometry with the data coming from an IMU sensor, in order to achieve environmental depth perception in real-time and ego-motion estimation on a hand-held device. Von Marcard et al. [29] proposed a method that combines a set of inertial measurement units attached to the body limbs and a single hand-held camera to compute precise three-dimensional poses in the wild. Trumble et al. [30] presented an algorithm for fusing IMU sensors data with multi-viewpoint video (MVV), with the aim of accurately estimating the three-dimensional human pose. They incorporated the pose embedding learnt from a 3D convolutional neural network with a forward kinematic solve of the inertial data and they found that the hybrid pose inference obtained from these two data sources can resolve the ambiguities of each sensor modality, yielding a better accuracy. Malleson et al. [31] designed a real-time full-body motion capture system which takes as inputs the data coming from a sparse set of IMUs and the images produced by two or more video cameras and uses a framework based on some optimization criterion to incorporate in real-time constraints set by the inertial sensors, by the cameras and by a prior pose model. In this way, they managed to recover the full 6-DOF motion, including the global positions free of any drift error, and they were able to track in real-time a broad variety of human motions in unconstrained indoor as well as outdoor settings. Von Marcard et al. [32] proposed a method to fuse video with sparse orientation data coming from a small number of IMU sensors to improve full-body human motion capture and perform human pose estimation. Their hybrid tracker is able to compensate for the drawbacks of each sensor type: it provides precise limb orientation and good results during rapid motions from inertial sensors and, at the same time, drift-free and accurate position information from video data.

1.3 Goals

Most of the literature cited so far shows qualitative 3D reconstruction results, very often convincing on the perceptual side but lacking in details (timing, accuracy) that could severely affect a cobotic application. For this reason, this thesis focuses on the integration of inertial measurement units and a stereo vision system in order to deal with the reconstruction problem in a more quantitative way. The final aim is to improve the human localization derived from each single system and to fuse both systems to produce measurable additional improvements, significant for collaborative robotics purposes. The proposed experiments are designed to evaluate the accuracy of the reconstruction framework. First, a data acquisition system composed of eight IMU sensors attached to the human body and two cameras is designed. Then, the IMU-based system and the camera-based system are singularly calibrated and evaluated. Finally, a two-step Kalman filter fusion algorithm is used to integrate the inertial and vision measurements and to reconstruct the upper body skeleton of a human operator in a robotic collaborative environment. The framework is validated by simple calibration movements and by some more complex movements, in order to test the accuracy of the algorithm and to verify if such a system can be effectively employed in collaborative robotics applications.

1.4 Thesis outline

The remaining chapters of this thesis are organized as follows. In chapter 2 the IMU sensors are introduced and the upper body reconstruction problem using inertial sensors measurements is formalized. Chapter 3 includes the definition of the camera model and a short description of the camera calibration method. The stereo vision system is thus described and the reconstruction of the human skeleton from vision data is formalized. Chapter 4 provides a brief introduction to the Kalman filter and gives details of the Kalman filter fusion algorithm adopted in this thesis. In chapter 5 the experiments instrumentation and setup are presented. Chapter 6 presents the results of the experimental activity; the analysis and the discussion of these results is left to chapter 7. Chapter 8 draws some conclusions to the work carried on in this thesis and details some potential applications of a system fusing IMUs and vision data.

Chapter 2

Inertial Measurement Units (IMUs)

An Inertial Measurement Unit or IMU is an electronic device equipped with accelerometers, gyroscopes and sometimes also magnetometers, able to measure orientation and acceleration. IMUs are small and integrated and, in contrast to cameras, they are able to provide direct three-dimensional measurements, they do not suffer from occlusions and their workspace is not restricted to a special room equipped with cameras. For these reasons, they are widely used as a wearable tool for human motion tracking, motion capture and motion evaluation. However, they suffer from some well-known limitations. Their main drawback is that they are affected by a consistent drift error and therefore they alone cannot provide a good accuracy, required for example during collaborative tasks. Moreover, in order to find a positional measurement from an IMU it is possible to derive the acceleration data, but this is often numerically unstable, and the orientation measurements suffer from temporal lag. Lastly, wearing many inertial sensors can feel intrusive and limit the range of motion of the subject.

Despite these limitations, it is still possible to reconstruct the human skeleton using inertial measurement units, as many researchers did in the past. The first operation to be executed when dealing with IMUs is their calibration, which will be explained in chapter 5, after the sensors used in this thesis work are introduced.

2.1 Human Skeleton Reconstruction from IMU Sensors: Problem Formalization

The approach followed for the mathematical formalization of the problem of reconstructing the human skeleton using inertial sensors is inspired by the work of Zhang [16] and Diebel [33].
Three different types of coordinate frames are needed to formulate the problem:

- one global reference system ($\boldsymbol{F}^g$), positioned on the ground with the positive z-azis pointing upwards, the positive x-axis pointing to the right side of the body and the positive y-axis pointing forward, according to the right hand rule

- one body reference system per each body segment ($\boldsymbol{F}^b$), centered on each joint and oriented exactly as the global reference system, according to the right hand rule
- one IMU reference system per each IMU sensor ($\boldsymbol{F}^i$), centered on each sensor.

The relationship between the three reference systems mentioned above, necessary to derive the 3D structure of the body, is shown in figure 2.1 and is illustrated in the following sections.
The different reference frames placed on the human body are shown in figure 2.2.

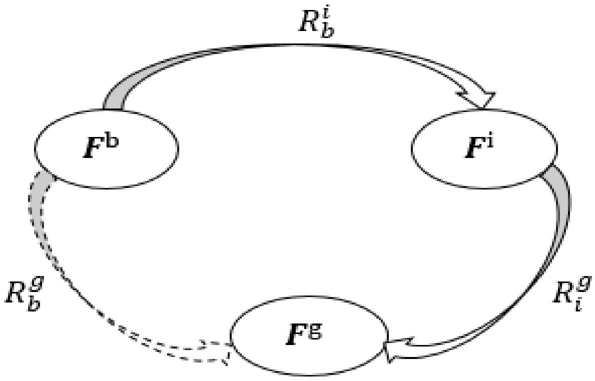


Figure 2.1: Relationship between the three reference systems $\boldsymbol{F}^g$, $\boldsymbol{F}^b$ and $\boldsymbol{F}^i$ for each body segment. The two continuous arrows indicate the two direct transforms between the body and the IMU reference frames (called alignment matrices in the following sections) and between the IMU and the global reference frames (called initial and instantaneous rotation matrices in the following sections). The dashed arrow indicates the indirect transform between the body and the global reference frames, which can be obtained by combining the two direct transforms.

2.1.1 Alignment Matrices

Sensor-to-body alignment (or anatomical calibration) is a procedure commonly adopted in order to precisely define the relation between sensors and body segments. It consists in aligning the sensor axes with the anatomical axes, by finding the rotation matrix between each IMU reference frame and the reference frame of the body segment to which the IMU is attached. This step is not compulsory, since the movement of the body segments will slightly modify the position of the sensors on the body, and therefore some researchers completely skip it [34][35]. However, it was demonstrated that this anatomical calibration yields a better overall performance in terms of measurement accuracy, reliability and repeatability [36][37], because the sensors may be attached to curved body surfaces or on active skeletal muscles and consequently it could be difficult to position the IMU sensors in such a way to guarantee a good alignment with anatomical segments. For this reason, other researchers tackle this problem using post-processing of standard motion tests data [36] or a deep learning approach [38].

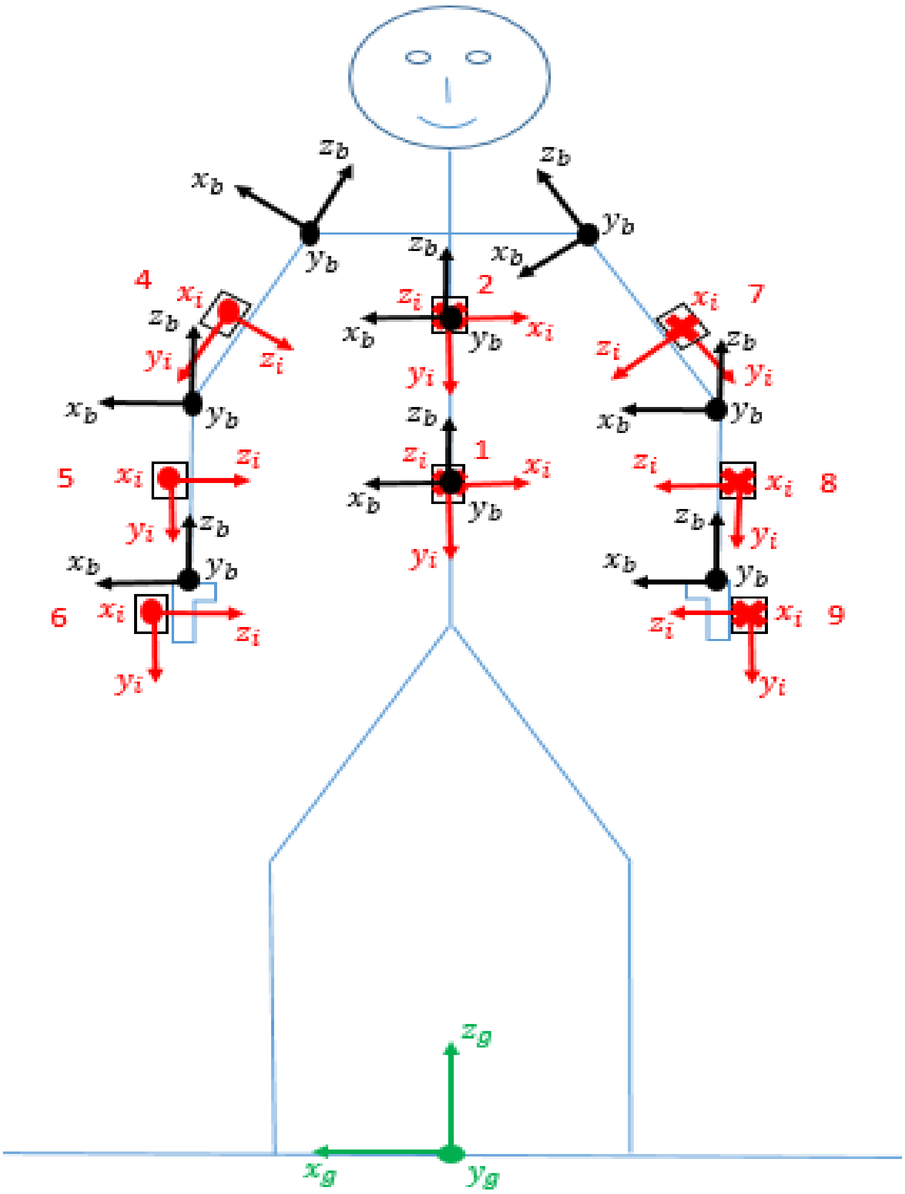


Figure 2.2: Global reference system (in green), eight body reference systems (in black) and eight IMU reference systems (in red) positioned on the human body. The eight IMU sensors are numbered in red.

For the purposes of this thesis, it is necessary to find eight rotation matrices $\mathbf{R}_b^i$ (one per IMU sensor) which report a vector expressed in the body reference system to the corresponding IMU reference system, according to the following equation (2.1):

$$\mathbf{v}_i = \mathbf{R}_b^i \cdot \mathbf{v}_b \tag{2.1}$$

To this aim, the anatomical calibration is performed using the data recorded by the inertial sensors positioned on the test subject's body while he is performing a few simple rotations of his right and left arms and of his torso along a known direction. In particular:

- for the arms, the subject first stands still for a few seconds in the neutral position, with the arms lowered down along the body (stationary phase). Then he performs a shoulder rotation in the sagittal plane, raising the right arm forward and lowering it down again. This rotation is repeated five times. Once this movement is over, he stands still for a few seconds in the neutral position and then he performs the same five rotations in the sagittal plane with his left arm. After standing still in the neutral position again for a few seconds, he performs a shoulder rotation in the coronal plane, raising the right arm sideways and lowering it down again for five times. Finally, he repeats the same five sideways rotations with his left arm.
  The difference between the sagittal, coronal and transverse planes is shown in figure 2.3. The inertial data recorded by the IMU number 6, positioned on the test subject's right hand, during the right arm calibration movements is shown in figure 2.4: the data from the sagittal rotation is reported in figure 2.4a, while the data from the coronal rotation is reported in figure 2.4b.

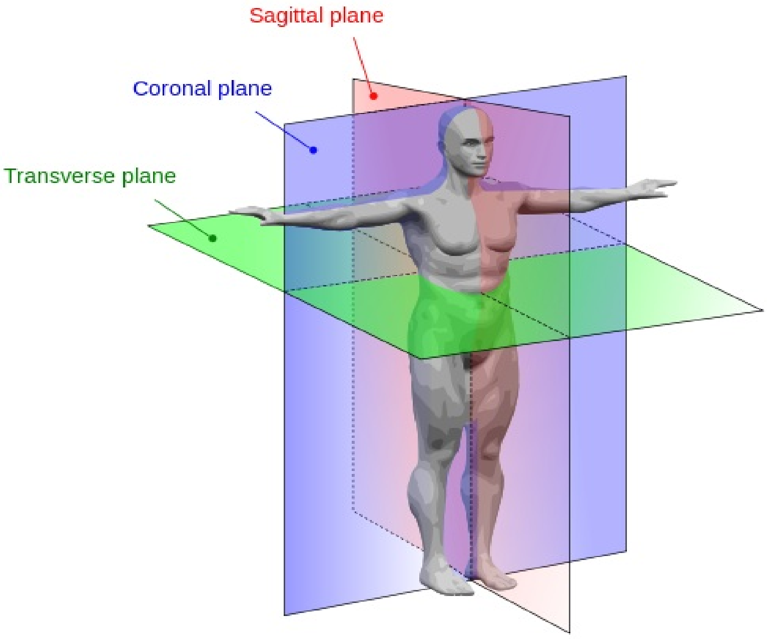
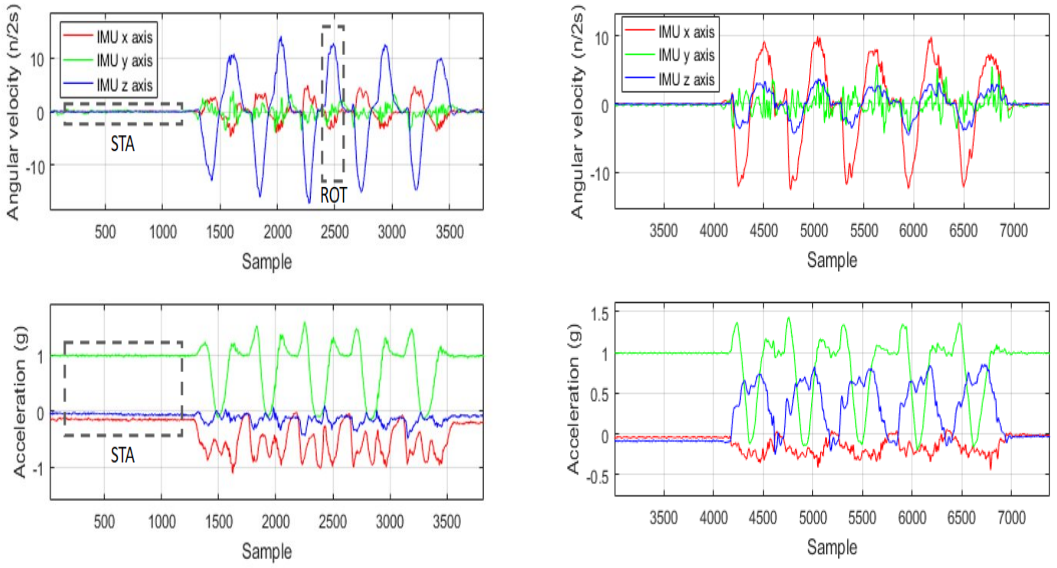


Figure 2.3: Sagittal, coronal and transverse planes.

- for the torso and neck, the subject first stands still for a few seconds in the neutral position, then he performs a torso rotation in the sagittal plane, lowering the upper body forward and rising up again. Also this rotation is repeated five times. The inertial data recorded by the IMU number 1, positioned on the test subject's torso, during the torso calibration movements is shown in figure 2.5.



(a) Inertial data recorded by the IMU sensor number 6, positioned on the subject's right hand, during the first calibration movement. Since the arm is rotated forward in the sagittal plane, the rotation (and therefore the angular velocity) is along the IMU z-axis (plotted in blue).

(b) Inertial data recorded by the IMU sensor number 6, positioned on the subject's right hand, during the second calibration movement. Since the arm is rotated sideways in the coronal plane, the rotation (and therefore the angular velocity) is along the IMU x-axis (plotted in red).

Figure 2.4: Right hand calibration movements. In the two top plots the output of the gyroscope, so the angular velocity expressed in semi-turns per second, on the three axes (x,y,z) is plotted in red, green and blue, respectively. In the two bottom plots the output of the accelerometer, so the acceleration expressed as a function of gravity g, on the three axes (x,y,z) is plotted in red, green and blue, respectively.

During the stationary phase (the one in the pane denoted as STA in figure 2.4a), the acceleration recorded by the IMU sensors in the z_b direction is the gravity vector. Then, the accelerometers readings averaged and normalized during the stationary period constitute the third column of the alignment matrix $\mathbf{R}_b^i$.

The gyroscope data in one of the two rotation directions (for instance the one in the pane denoted as ROT in figure 2.4a), integrated with respect to time and normalized, are instead the vector $\mathbf{c}_1^t$, which has the same direction of the first column of $\mathbf{R}_b^i$. The first and second columns of each alignment matrix $\mathbf{R}_b^i$ are then found according to equations 2.2 and 2.3:

$$\mathbf{c}_2 = \frac{\mathbf{c}_3 \times \mathbf{c}_1^t}{||\mathbf{c}_3 \times \mathbf{c}_1^t||} \tag{2.2}$$

$$\mathbf{c}_1 = \frac{\mathbf{c}_2 \times \mathbf{c}_3}{||\mathbf{c}_2 \times \mathbf{c}_3||} \tag{2.3}$$

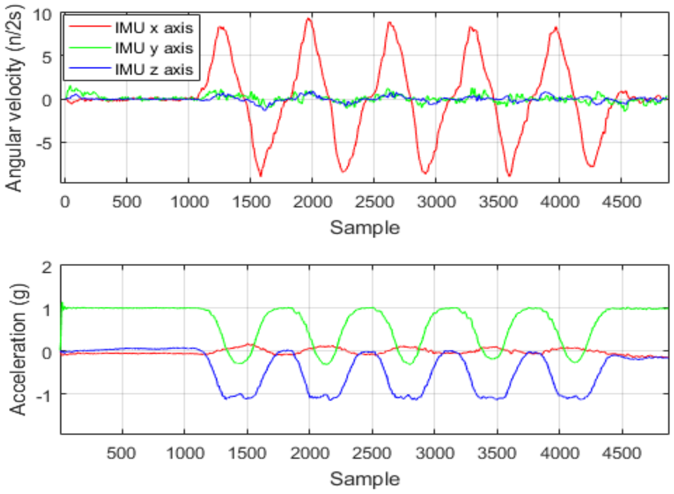


Figure 2.5: Torso calibration movements. The rotation is around the IMU 1 x-axis (shown in red).

Once the alignment matrices from the body reference frames to the IMU reference frames have been determined, it is necessary to find the coordinate transformations between the global reference frame and each IMU reference frame. To do so, the initial rotation matrices $\mathbf{R}_{0i}^{g}$ and the instantaneous rotation matrices $\mathbf{R}_{i}^{g}$ need to be defined.

2.1.2 Initial Rotation Matrices

The initial rotation matrix represents the initial attitude of each IMU sensor in the global reference frame. It can be found from the corresponding initial quaternion, which in turn is determined from the Euler angles sequence (ϕ, θ, ψ) relating the initial attitude of each IMU reference frame with the global reference frame. The detailed procedure is explained in the following sections.

2.1.2.1 The Euler angles

The Euler angles are three angles introduced by Leonhard Euler to represent the orientation of a rigid body with respect to a fixed coordinate system. Euler angles are typically denoted as (ϕ, θ, ψ) and called respectively roll, pitch and yaw. These terms define a sequence of three elementary rotations, which can be executed in different combinations. One of the most used sequences is the *z-y-x* or *3-2-1* sequence. Considering this sequence, the three orthogonal matrices associated to the rotations (ψ, θ, ϕ) are:

- the rotation of an angle ϕ along the z-axis of the fixed reference frame (equation 2.4):

$$\mathbf{R}_3(\psi) = \begin{bmatrix} c(\psi) & -s(\psi) & 0 \\ s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

- the rotation of an angle θ along the axis of the intermediate reference frame y_1 (equation 2.5):

$$\mathbf{R}_2(\theta) = \begin{bmatrix} c(\theta) & 0 & s(\theta) \\ 0 & 1 & 0 \\ -s(\theta) & 0 & c(\theta) \end{bmatrix} \tag{2.5}$$

- the rotation of an angle ψ along the axis of the intermediate reference frame x_2 (equation 2.6):

$$\mathbf{R}_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi) & c(\phi) \end{bmatrix} \tag{2.6}$$

The rotation matrix allowing to change from the each IMU reference frame to the global reference frame is obtained multiplying the three aforementioned matrices:

$$\mathbf{R}_i^g = \mathbf{R}_1(\phi) \cdot \mathbf{R}_2(\theta) \cdot \mathbf{R}_3(\psi) \tag{2.7}$$

which in extended form becomes:

$$\mathbf{R}_i^g = \begin{bmatrix} c(\psi)c(\theta) & c(\theta)s(\psi) & -s(\theta) \\ c(\psi)s(\phi)s(\theta) - c(\phi)s(\psi) & c(\phi)c(\psi) + s(\phi)s(\psi)s(\theta) & c(\theta)s(\phi) \\ s(\phi)s(\psi) + c(\phi)c(\psi)s(\theta) & c(\phi)s(\psi)s(\theta) - c(\psi)s(\phi) & c(\phi)c(\theta) \end{bmatrix} \tag{2.8}$$

The three initial Euler angles (i.e. corresponding to the neutral or rest position) can be found using equation 2.9 for the IMUs on the right arm, equation 2.10 for the IMUs on the left arm and equation 2.11 for the IMUs on the torso and on the neck:

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} atan2(r_2, r_3) \\ -asin(r_1) \\ \frac{\pi}{2} \end{bmatrix} \tag{2.9}$$

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} atan2(r_2, r_3) \\ -asin(r_1) \\ -\frac{\pi}{2} \end{bmatrix} \tag{2.10}$$

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} atan2(r_2, r_3) \\ -asin(r_1) \\ \pi \end{bmatrix} \tag{2.11}$$

where r_1, r_2 and r_3 are the accelerations along the x, y and z axis respectively, registered by each IMU during the stationary period, expressed in the IMU reference frame and normalized with respect to the gravitational force, and *atan2* is the four quadrant inverse tangent function.

The Euler angles representation of the orientation is easily interpretable, but little efficient and computationally expensive, due to the presence of many trigonometric functions and the possibility to have numeric singularities. Therefore, in order to represent the orientation and to calculate the attitude changes in time it is more efficient and computationally more advantageous to use quaternions.

2.1.2.2 The quaternions

The quaternions are mathematical entities which were first introduced in 1843 by Irish mathematician William Rowan Hamilton as an extension to complex numbers and they are applied still today to represent a body attitude in the three-dimensional space. By defining an axis of instantaneous rotation with a versor:

$$\hat{\mathbf{u}} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}$$

and the rotation around such axis of an angle θ, the corresponding quaternion can be defined as follows (equation 2.12):

$$\mathbf{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} u_x \cdot sin(\frac{\theta}{2}) \\ u_y \cdot sin(\frac{\theta}{2}) \\ u_z \cdot sin(\frac{\theta}{2}) \\ cos(\frac{\theta}{2}) \end{bmatrix} \tag{2.12}$$

where q_0 is the scalar part of the quaternion and q_1, q_2 and q_3 are the three components of the vectorial part of the quaternion. This quaternion represents a rotation with respect to the unit direction vector $\hat{\mathbf{u}}$ through the angle θ.
The adjoint, the norm and the inverse of quaternion $\mathbf{q}$ are:

$$\bar{\mathbf{q}} = \begin{bmatrix} q_0 \\ -q_1 \\ -q_2 \\ -q_3 \end{bmatrix} \qquad ||\mathbf{q}|| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \qquad \mathbf{q}^{-1} = \frac{\bar{\mathbf{q}}}{||\mathbf{q}||}$$

One important property of quaternions is that quaternion multiplication is not commutative.

Quaternions are used in pure and applied mathematics, in particular when there is the necessity to perform calculations involving three-dimensional rotations such as in 3D computer graphics and computer vision. In practical applications, for example when it is required to determine the spatial orientation of a body, they can be used as an alternative to other methods like Euler angles and rotation matrices, since their algebra is easier and they are computationally much more efficient. In order to represent the attitude of a rigid body, a quaternion must have unitary norm:

$$||\mathbf{q}|| = 1$$

For the purposes of this thesis, using the three Euler angles calculated in the previous section it is easy to find the initial quaternion $\mathbf{q}_0$ from equation 2.13 and the corresponding initial rotation matrix $\mathbf{R}_{0i}^{g}$ from equation 2.14:

$$\mathbf{q}_0 = \begin{bmatrix} c_{\phi/2}c_{\theta/2}c_{\psi/2} + s_{\phi/2}s_{\theta/2}s_{\psi/2} \\ -c_{\phi/2}s_{\theta/2}s_{\psi/2} + s_{\phi/2}c_{\theta/2}c_{\psi/2} \\ c_{\phi/2}s_{\theta/2}c_{\psi/2} + s_{\phi/2}c_{\theta/2}s_{\psi/2} \\ c_{\phi/2}c_{\theta/2}s_{\psi/2} - s_{\phi/2}s_{\theta/2}c_{\psi/2} \end{bmatrix} \tag{2.13}$$

$$\mathbf{R}_{0i}^{g} = \begin{bmatrix} q_{0,0}^2 + q_{0,1}^2 - q_{0,2}^2 - q_{0,3}^2 & 2(q_{0,1}q_{0,2} + q_{0,0}q_{0,3}) & 2(q_{0,1}q_{0,3} - q_{0,0}q_{0,2}) \\ 2(q_{0,1}q_{0,2} - q_{0,0}q_{0,3}) & q_{0,0}^2 - q_{0,1}^2 + q_{0,2}^2 - q_{0,3}^2 & 2(q_{0,2}q_{0,3} + q_{0,0}q_{0,1}) \\ 2(q_{0,1}q_{0,3} + q_{0,0}q_{0,2}) & 2(q_{0,2}q_{0,3} - q_{0,0}q_{0,1}) & q_{0,0}^2 - q_{0,1}^2 - q_{0,2}^2 + q_{0,3}^2 \end{bmatrix} \tag{2.14}$$

2.1.3 Instantaneous Rotation Matrices

When each IMU sensor moves, the corresponding quaternion changes and it can be updated using the quaternion update found from equation 2.15:

$$\frac{d}{dt}\mathbf{q} = \frac{1}{2}\Omega\left[\boldsymbol{\omega}^i\right] \cdot \mathbf{q} \tag{2.15}$$

where $\mathbf{q}$ is the quaternion at the previous time instant and the 4×4 matrix $\Omega\left[\boldsymbol{\omega}^i\right]$ is constructed using the angular velocity vector expressed in the IMU reference frame, so the output of the gyroscope $\boldsymbol{\omega}^i$, according to equation 2.16:

$$\Omega\left[\boldsymbol{\omega}^i\right] = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \tag{2.16}$$

After the updating process, the new quaternion must be forced to have unitary norm, using a practical solution (equation 2.17):

$$\boldsymbol{q}^+ = \frac{\boldsymbol{q}}{||\boldsymbol{q}||} \tag{2.17}$$

where $\boldsymbol{q}^+$ is the new quaternion with unitary norm.
$\boldsymbol{q}^+$ can be used at every sample instant to calculate the instantaneous rotation matrix $\mathbf{R}_i^g$ from each IMU sensor reference system to the global reference system, following equation 2.14.

2.1.4 Upper Body Trajectory Reconstruction

While each IMU sensor moves, the accelerometer inside it registers the combination of two different accelerations: the gravitational acceleration and the linear acceleration. The latter represents the real movement of the body segment to which the inertial sensor is attached. At each time step the total acceleration can be reported from the IMU reference frame to the global reference frame using the matrix $\mathbf{R}_i^g$ just found, and then the gravitational acceleration $\mathbf{g}$ expressed in the global reference frame can be removed in order to get the linear acceleration of the inertial sensor in the global reference frame $\boldsymbol{a}^g$ (equation 2.18):

$$\boldsymbol{a}^g = \mathbf{R}_i^g \cdot \boldsymbol{a}^i - \boldsymbol{g}^g \tag{2.18}$$

where $\boldsymbol{g}^g = \begin{bmatrix} 0 & 0 & |g| \end{bmatrix}^T$.

The acceleration can then be integrated once or twice with respect to time in order to find respectively the linear velocity and the position of each IMU sensor in the global reference frame (equations 2.19 and 2.20):

$$\boldsymbol{v}^g = \int \boldsymbol{a}^g \cdot dt \tag{2.19}$$

$$\boldsymbol{s}^g = \int\int \boldsymbol{a}^g \cdot dt^2 \tag{2.20}$$

Using the initial and instantaneous rotation matrices found in the previous sections, it is possible to determine the positions of each joint in the global reference system at every time step. In doing so, all upper body segments are assumed to behave as rigid bodies.
First, it is necessary to define the position of each joint with respect to the corresponding IMU system in the body reference system. For example, the right shoulder position with respect to the IMU number 4 attached to the right upper arm in the right upper arm reference system can be found as:

$$\mathbf{l}^{rua}_{rs/IMU4} = \begin{bmatrix} 0 \\ 0 \\ 20 \end{bmatrix}$$

and the right shoulder position with respect to the IMU number 2 attached to the neck in the neck reference system can be found as:

$$\mathbf{l}^{neck}_{rs/IMU2} = \begin{bmatrix} 20 \\ 0 \\ 6 \end{bmatrix}$$

where all lengths are expressed in centimeters.
The instantaneous position of the right shoulder joint in the global reference system can be computed from equations 2.21 and 2.22:

$$\mathbf{s}^g_{rs/IMU4} = \mathbf{R}^g_{IMU4}\mathbf{R}^{IMU4}_{rua} \cdot \mathbf{l}^{rua}_{rs/IMU4} + \mathbf{s4}^g \tag{2.21}$$

$$\mathbf{s}^g_{rs/IMU2} = \mathbf{R}^g_{IMU2}\mathbf{R}^{IMU2}_{neck} \cdot \mathbf{l}^{neck}_{rs/IMU2} + \mathbf{s2}^g \tag{2.22}$$

where $\mathbf{R}^{IMU4}_{rua}$ and $\mathbf{R}^{IMU2}_{neck}$ are respectively the alignment matrix from the right upper arm reference frame to the IMU number 4 reference frame and the alignment matrix from the neck reference frame to the IMU number 2 reference frame, $\mathbf{R}^g_{IMU4}$ and $\mathbf{R}^g_{IMU2}$ are either the initial rotation matrices or the instantaneous rotation matrices (depending if we are considering the first time step or any following time step) for IMU number 4 and for IMU number 2 respectively, and $\mathbf{s4}^g$ and $\mathbf{s2}^g$ are the positions of IMU number 4 and of IMU number 2 in the global reference system, which at the beginning are set according to the predefined global reference system shown in figure 2.2, and at the subsequent time steps are updated with the result of the integration in equation 2.20.

Following the same procedure, it is possible to determine the positions of all eight upper body joints in the global reference system at every time step.

2.1.4.1 Anatomical Constraints

In order to ensure that the joints do not break, some anatomical constraints need to be introduced. For the first body segment of the kinematic chain, which connects the torso joint to the neck joint, a vector **v** is defined which represents the torso displacement in the last time step (equation 2.23):

$$\boldsymbol{v} = \boldsymbol{s}_{torso}(t-1) - \boldsymbol{s}_{torso}(t) \tag{2.23}$$

For the following body segments of the kinematic chain, vector **v** connects instead the position of each joint calculated from the following IMU sensor in the kinematic chain to the position of the same joint calculated from the previous IMU sensor in the kinematic chain, to ensure that two consecutive body segments remain always attached during motion. For example, the displacement of the neck joint during the last time step is computed as the difference between the neck position calculated from IMU number 1 (placed on the torso) at time t and the neck position calculated from IMU number 2 (placed on the neck) at time t, as reported in equation 2.24:

$$\boldsymbol{v} = \boldsymbol{s}_{neck/IMU1}(t) - \boldsymbol{s}_{neck/IMU2}(t) \tag{2.24}$$

Vectors **v** are used to update the positions of the two joints located at the two extremities of each body segment, which were previously calculated following the same procedure as in equations 2.21 and 2.22. For example, the vector **v** computed in equation 2.23 will be used to update the positions of the torso and neck joints.

Furthermore, in order to avoid any integration errors which would cause a big drift in the x, y and z directions in the upper body reconstruction, at every time step the position of each IMU sensor is corrected using the same vector **v** and the velocity is recomputed by deriving the position with respect to time.

Chapter 3

Cameras

The content of the first sections of this chapter is inspired by the work of Olivier Faugeras [39].

A camera is an optical instrument used to capture images. Cameras are basically sealed boxes with a small hole, called aperture, that lets light in to record an image on a light-sensitive surface, which is usually photographic film or a digital sensor. The aperture can be enlarged or narrowed to let more or less light into the camera and the lenses can focus the light entering through the aperture. At the end of the day, a camera functions in a very similar way as the human eye.

3.1 Camera model

The most commonly used model for a camera is the pinhole model. It consists of two screens: on the first screen a hole has been punched, such that the rays of light emitted or reflected by an object can pass, forming an inverted image of the object on the second screen. As shown in figure 3.1, the image **p** of a 3D point **P** is formed on the image plane as the intersection between the line connecting the point **P** and the optical center (or center of projection) of the camera with the image plane. This operation is called *perspective projection*. The plane passing through the optical center and parallel to the image plane is called focal plane: the focal length is the distance f between the image plane and the focal plane. The line passing through the optical center and perpendicular to the image plane is referred to as the optical axis.

Two coordinate systems can be defined:

- the coordinate system *(X,Y,Z)* for the 3D space
- the coordinate system *(u,v)* for the image plane

as indicated in figure 3.2.
(u_0,v_0) are the coordinates of the center of the image plane.

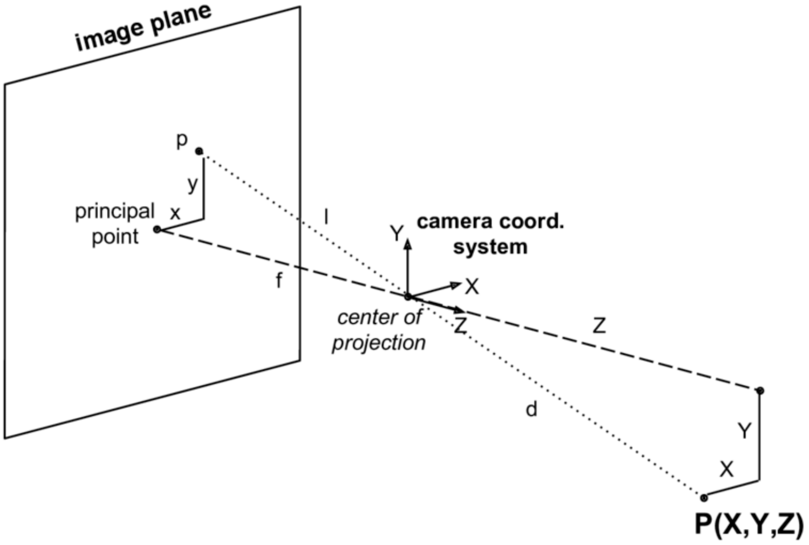


Figure 3.1: Pinhole camera model and perspective projection.

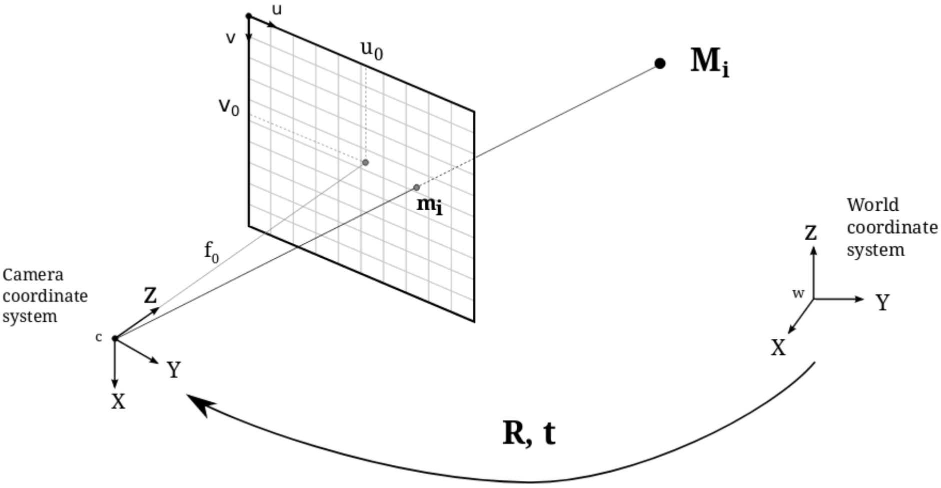


Figure 3.2: Pinhole camera coordinate systems.

The relationship between a point in 3D coordinates and its projection on the image coordinates can be expressed as (equation 3.1):

$$-\frac{f}{z} = \frac{u}{x} = \frac{v}{y} \tag{3.1}$$

which can be rewritten linearly as (equation 3.2):

$$\begin{bmatrix} U \\ V \\ S \end{bmatrix} = \begin{bmatrix} -f & 0 & 0 & 0 \\ 0 & -f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{3.2}$$

where

$$u = \frac{U}{S} \qquad v = \frac{V}{S} \qquad if \quad S \neq 0$$

Equation 3.2 is projective, which means that it is defined up to a scale factor S. It is possible to rewrite it using the projective coordinates (X, Y, Z, T) of the 3D point **P** (equation 3.3):

$$\begin{bmatrix} U \\ V \\ S \end{bmatrix} = \begin{bmatrix} -f & 0 & 0 & 0 \\ 0 & -f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ T \end{bmatrix} \tag{3.3}$$

The above equation shows that the relationship between image coordinates and space coordinated is linear in projective coordinates. In matrix form (equation 3.4):

$$\mathbf{p} = \mathbf{M} \cdot \mathbf{P} \tag{3.4}$$

where

$$\mathbf{p} = [U, V, S]^T \qquad and \qquad \mathbf{P} = [X, Y, Z, T]^T$$

3.2 Camera calibration

The first task to be performed when dealing with cameras is camera calibration, a necessary step in order to extract three-dimensional information from two-dimensional images. In general, the problem consists of two steps:

- estimating the perspective projection matrix **M**, a 3 × 4 matrix which describes the mapping of a pinhole camera from 3D points in the world coordinate system (x_w, y_w, z_w) to 2D points in an image coordinate system (u,v)
- estimating from **M** the intrinsic and extrinsic camera parameters, expressed by the matrices **A** and (**R**,**t**), respectively.

For some applications, for example for stereo vision, the second step may not be necessary. The relationship between the different coordinate systems is displayed in figure 3.3.

3.2.1 Intrinsic parameters

The most general matrix **M** can be written as (equation 3.5):

$$\mathbf{M} = \begin{bmatrix} -f \cdot k_u & 0 & u_0 & 0 \\ 0 & -f \cdot k_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.5}$$

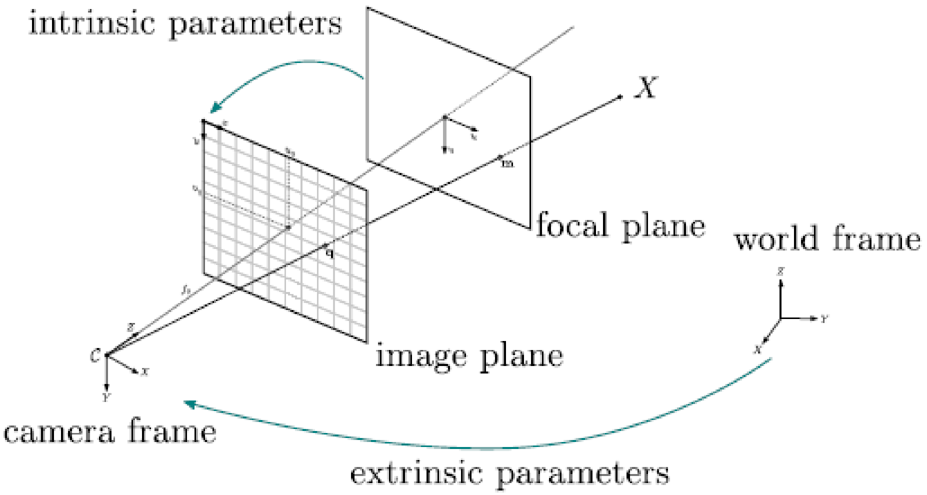


Figure 3.3: Different coordinate systems and intrinsic and extrinsic parameters.

where f is the focal length, k_u and k_v are two coefficients whose interpretation will be given in the following paragraphs and u_0 and v_0 are the coordinates of the intersection of the optical axis with image plane, so the coordinates of the center of the image plane. By letting $\alpha_u = -f \cdot k_u$ and $\alpha_v = -f \cdot k_v$, matrix $\mathbf{M}$ becomes (equation 3.6):

$$\mathbf{M} = \begin{bmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.6}$$

The scale factors α_u and α_v and the coordinates u_0 and v_0 do not depend on the spatial position and orientation of the camera, and therefore they are called *intrinsic parameters.*

If we model the camera using the pinhole model, the projection equation relating a point seen on the image plane and the same point in the camera frame is reported in equation 3.7:

$$\begin{bmatrix} U \\ V \\ S \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \tag{3.7}$$

where, if $S \neq 0$, $u = \frac{U}{S}$ and $v = \frac{V}{S}$ and x_c, y_c and z_c are the camera frame coordinates. Expressing x, y, z and f in units of length and u and v in pixel units, from equations 3.8 and 3.9 it is possible to interpret the meaning of the intrinsic parameters.

$$u = \frac{U}{S} = -f \cdot k_u \cdot \frac{x}{z} + u_0 \tag{3.8}$$

$$v = \frac{V}{S} = -f \cdot k_v \cdot \frac{y}{z} + v_0 \tag{3.9}$$

The quantities $\frac{1}{k_u}$ and $\frac{1}{k_v}$ can be interpreted as the size of the horizontal and vertical pixels in meters, respectively, while the parameters α_u and α_v can be interpreted as the size of the focal length in horizontal and vertical pixels, respectively.

3.2.2 Extrinsic parameters

The relationship between a point in the camera frame (expressed in the x_c, y_c and z_c coordinates) and the same point in the world frame (expressed in the x_w, y_w and z_w coordinates) is reported in equation 3.10:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{r_1} & \boldsymbol{r_2} & \boldsymbol{r_3} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ \mathbf{0_3}^T & 1 \end{bmatrix} \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = (\mathbf{R}, \boldsymbol{t}) \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3.10}$$

where the rotation matrix $\mathbf{R}$ and the translational vector $\mathbf{t}$ describe the position and orientation of the camera frame with respect to the world coordinate system. The three parameters which define $\mathbf{R}$ and the three parameters of $\mathbf{t}$ are called the *extrinsic parameters* of the camera.

3.2.3 Estimating M

The general form of the perspective projection matrix $\mathbf{M}$, written as a function of the intrinsic and extrinsic parameters, is reported in equation 3.11:

$$\mathbf{M} = \begin{bmatrix} \alpha_u \cdot \boldsymbol{r}_1 + u_0 \cdot \boldsymbol{r}_3 & \alpha_u \cdot t_x + u_0 \cdot t_z \\ \alpha_v \cdot \boldsymbol{r}_2 + v_0 \cdot \boldsymbol{r}_3 & \alpha_v \cdot t_y + v_0 \cdot t_z \\ \boldsymbol{r}_3 & t_z \end{bmatrix} \tag{3.11}$$

where the vectors $\boldsymbol{r}_1$, $\boldsymbol{r}_2$ and $\boldsymbol{r}_3$ are the row vectors of matrix $\mathbf{R}$ and t_x, t_y and t_z are the three components of the translation vector $\boldsymbol{t}$ in the x, y and z directions, respectively.

In total, there are four intrinsic parameters (the scale factors α_u and α_v and the coordinates u_0 and v_0) and six extrinsic parameters (three for the rotation and three for the translation from the world coordinates system to the camera coordinate system) to be determined. As mentioned at the beginning of this section, for stereo vision it is possible to calibrate the camera estimating only the perspective projection matrix $\mathbf{M}$, without estimating its intrinsic and extrinsic parameters. To do so, it is enough to combine equations 3.7 and 3.10 in order to relate a point coordinates on the image plane directly to its coordinates in the world reference system, through the perspective projection matrix $\mathbf{M}$. This step is reported in equation 3.12:

$$s \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} U \\ V \\ S \end{bmatrix} = \mathbf{A} \cdot (\mathbf{R}, \boldsymbol{t}) \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = \mathbf{M} \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3.12}$$

In order to perform camera calibration, it is necessary to estimate all 12 parameters of the matrix **M**.
By developing equation 3.7, it results that one point of an image gives us the two following equations (3.13):

$$\begin{cases} u = \frac{m_{11} \cdot x_w + m_{12} \cdot y_w + m_{13} \cdot z_w + m_{14}}{m_{31} \cdot x_w + m_{32} \cdot y_w + m_{33} \cdot z_w + m_{34}} \\ v = \frac{m_{21} \cdot x_w + m_{22} \cdot y_w + m_{23} \cdot z_w + m_{24}}{m_{31} \cdot x_w + m_{32} \cdot y_w + m_{33} \cdot z_w + m_{34}} \end{cases} \tag{3.13}$$

Since the 3×4 matrix **M** is defined up to a scale factor, it is possible to divide everything by one of the parameters, for example by m_{34}, by assuming that it is equal to 1. With a little bit of algebra, it is easy to reformulate the latter equations as follows (3.14):

$$\begin{cases} -m_{11} \cdot x_w - m_{12} \cdot y_w - m_{13} \cdot z_w - m_{14} + m_{31} \cdot x_w \cdot u + m_{32} \cdot y_w \cdot u + m_{33} \cdot z_w \cdot u = -u \\ -m_{21} \cdot x_w - m_{22} \cdot y_w - m_{23} \cdot z_w - m_{24} + m_{31} \cdot x_w \cdot v + m_{32} \cdot y_w \cdot v + m_{33} \cdot z_w \cdot v = -v \end{cases} \tag{3.14}$$

The system has now 2 equations and 11 unknowns. It is solvable using the Least Squares method with at least 6 points, by solving the following system (3.15)):

$$\mathbf{Q} \cdot \boldsymbol{b} = \boldsymbol{d} \tag{3.15}$$

where matrix **Q** contains all the known parameters (x_w, y_w, z_w, u and v) of the considered points, vector ***b*** contains all the unknowns (m_{11}, m_{12},..., m_{33}) and vector ***d*** contains the known (u,v) of the different points.
The solution is obtained using the following equation (3.16):

$$\boldsymbol{b} = (\mathbf{Q}^T \cdot \mathbf{Q})^{-1} \cdot \mathbf{Q}^T \cdot \boldsymbol{d} \tag{3.16}$$

Once the system is solved, all the parameters of the perspective projection matrix **M** are determined. As hinted before, since this thesis uses a stereo vision application, this solution does not procure the camera's intrinsic and extrinsic parameters. Therefore, for the purpose of this work the camera calibration is completed.

3.3 Stereo vision

A stereo vision system consists of two pinhole cameras which form two images (u_l,v_l) and (u_r,v_r) of the same point $\mathbf{P} = (x_w, y_w, z_w)$, expressed in the world reference frame. This configuration, shown in figure 3.4, allows the vision system to simulate human binocular vision, and therefore to capture three-dimensional images.

Given the two images on the two cameras' image planes, two problems arise:

- the *correspondence* problem: given a point (u_l,v_l) on the left camera's image plane, define to which point (u_r,v_r) on the right camera's image plane it corresponds to. The correspondence of two points means that they are the two images of the same 3D point **P**

- the *reconstruction* or *triangulation* problem: given two projections (u_l,v_l) and (u_r,v_r) of a point $\mathbf{P} = (x_w, y_w, z_w)$ on two images, determine the 3D coordinates of $\mathbf{P}$ in the world reference frame.

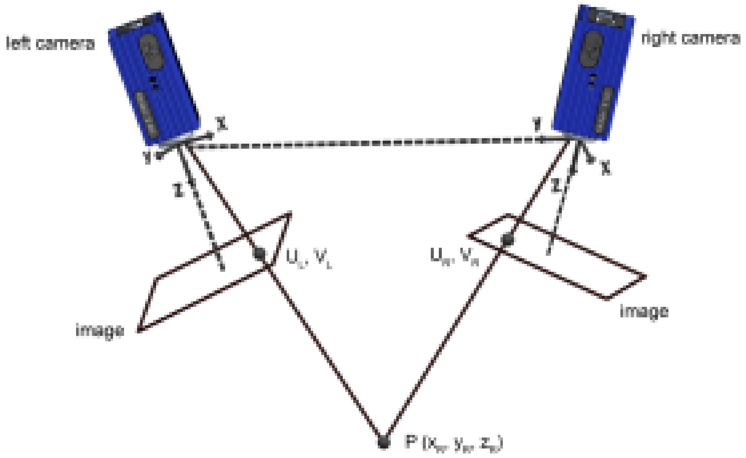


Figure 3.4: Stereoscopic system formed by two cameras.

3.3.1 The correspondence problem

The correspondence problem is always ambiguous, so there are some geometric and physical constraints that can be imposed to reduce the number of potential matches for a given point (u_l,v_l) on the left image.

3.3.1.1 The epipolar constraint

The first and most important constraint that can be imposed is the *epipolar* constraint, which arises from the geometry of stereo vision. From figure 3.5, it is clear that the point $\mathbf{P}$ that has produced the image p_l on the left image plane must lie on the half-line connecting the optical center of the left camera O_l and the projection p_l itself. Consequently, all possible matches p_r of p_l on the right image plane must lie on the image of this half-line, which is another half-line connecting p_r to the point e_r. e_r is the intersection between the line connecting the two optical centers of the two cameras O_l and O_r, whose length is referred to as baseline, and the right camera's image plane. e_r is called the epipole of the right camera with respect to the left one and the line connecting e_r and p_r is called the epipolar line of point p_l in the image plane of the right camera. Since the epipolar constraint is symmetric, also the possible matches for a point p_r in the right camera's image plane lie on the epipolar line through the epipole e_l, which is the intersection between the line connecting the two optical centers of the two cameras O_l and O_r and the left image plane. The two epipolar lines are the intersections of the epipolar plane O_lPO_r with the two cameras' image planes. In conclusion, each epipolar constraint states that for a given point p_l or p_r, on the left or right image plane respectively, all the possible matches in the other image plane lie on a line called the epipolar line. In this way it is possible to reduce the dimensions of the search space from two dimensions to one.

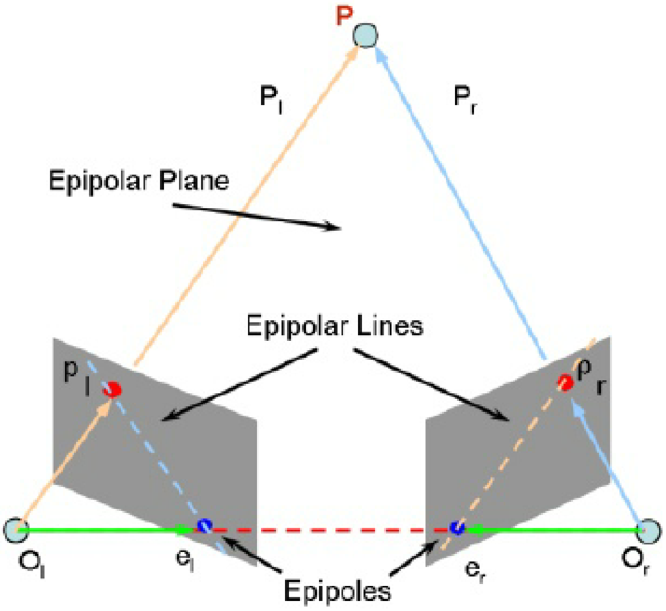


Figure 3.5: The epipolar geometry.

3.3.1.2 Other constraints

Other constraints that can be imposed in order to solve the correspondence problem include:

- *Uniqueness*: for opaque objects, one point oon the left image should have only one matching point on the right image. This constraint does not hold for transparent objects.

- *Continuity*: this constraint is based on the idea that most of the objects in the world have smooth surfaces. Let **P** be a 3D point with projections $p_l = (u_l, v_l)$ on the left image plane and $p_r = (u_r, v_r)$ on the right image plane. The *disparity* d is defined as the difference $d = v_r - v_l$. Then a neighbour n_l of p_l in the left image plane should have a match n_r on the right image plane with a disparity close to d.

- *Ordering*: objects in the world are usually bounded by continuous opaque surfaces. If we assume that the observed feature points lie on such a surface so as to be simultaneously visible to both image planes, we arrive at the ordering constraint: points on the same epipolar line are in the same order in both image planes' views. Therefore the ordering of edges or other features is usually preserved by stereo projection along epipolar lines. This means that if feature A is on the left of feature B in the left stereo image, the same spatial configuration is preserved in the right stereo image. More practically, the forbidden zone associated with a point of the surface is the cone defined by the point $\mathbf{P}$ itself and the two optical centers of the two cameras O_l and O_r. Any point belonging to this region has projections on the left and right image planes which violate the ordering constraint relative to point $\mathbf{P}$. Since it is easy to check if a point belongs to the forbidden zone depicted by point $\mathbf{P}$ considering the order of their images along the epipolar lines, this constraint can be used to eliminate matches for the point in the forbidden zone on one of the two image planes, given the match (p_l,p_r).

- *The disparity gradient*: let us consider a virtual retina parallel to the two real ones, called the cyclopean retina. If a point $\mathbf{P}$ has projections p_l and p_r on the two real retinas with coordinates v_l and v_r from their respective optical centers O_l and O_r, then its image $\mathbf{p}$ on the virtual retina has coordinates $\frac{v_l+v_r}{2}$ and the disparity d is defined as a smooth function of $w = \frac{v_l+v_r}{2}$. Let us consider two points on an object with cyclopean coordinates w_l and w_r and disparities d_l and d_r. The disparity gradient is defined as the magnitude of the derivative of the disparity with respect to the cyclopean coordinate: $DG =\mid \dfrac{d_1 - d_2}{w_1 - w_2} \mid$. The disparity gradient is upper-bounded: $DG < K$. A limit K of less than 2 implies that the matches between the two images preserve the topology of the images.

- *Geometric constraints*: this constraints restrict objects to be locally planar. For example, let us observe a curve C with three cameras and let us consider a point $\mathbf{P}$ on the curve. From two images p_1 and p_2 of point $\mathbf{P}$ it is possible to reconstruct $\mathbf{P}$ and therefore reproject it to predict p_3. In the same way, from the tangents $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ at p_1 and p_2 it is possible to predict the tangent $\boldsymbol{t}_3$ at p_3 and from the curvatures k_1 and k_2 at p_1 and p_2 it is possible to predict the curvature k_3 at p_3.

3.3.1.3 OpenPose

In this thesis, the correspondence problem was solved by means of the tool OpenPose. OpenPose is a real-time multi-person system that can detect human body, hand, facial, and foot keypoints (in total 135 keypoints) simultaneously on single images [40]. It has two main funcionalities:

- 2D real-time multi-person keypoint detection: 15 or 18 or 25-keypoint body or foot estimation, 6-keypoint foot estimation, 2x21-keypoint hand estimation and 70-keypoint face estimation.

- 3D real-time single-person keypoint detection: three-dimensional triangulation from multiple single views.

The program takes as input an image, a video, a webcam, a Flir/Point Grey or an IP camera and returns as output a basic image plus the keypoint display/saving in image or video format, the keypoint saving in text format (for example as a JSON file) and/or the keypoints as an array class. It runs on different operating systems, such as Ubuntu (14, 16), Windows (8, 10), Mac OSX and Nvidia TX2. Several versions are available for free download online, including the CUDA (Nvidia GPU), the OpenCL (AMD GPU) and the CPU-only (no GPU) versions.

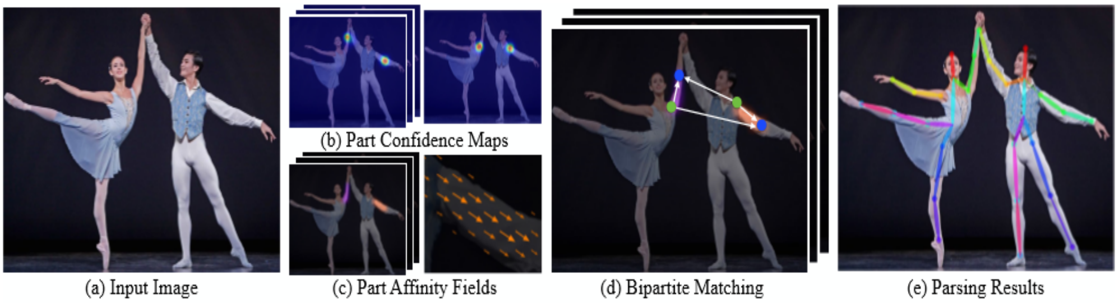


Figure 3.6: The overall OpenPose pipeline.

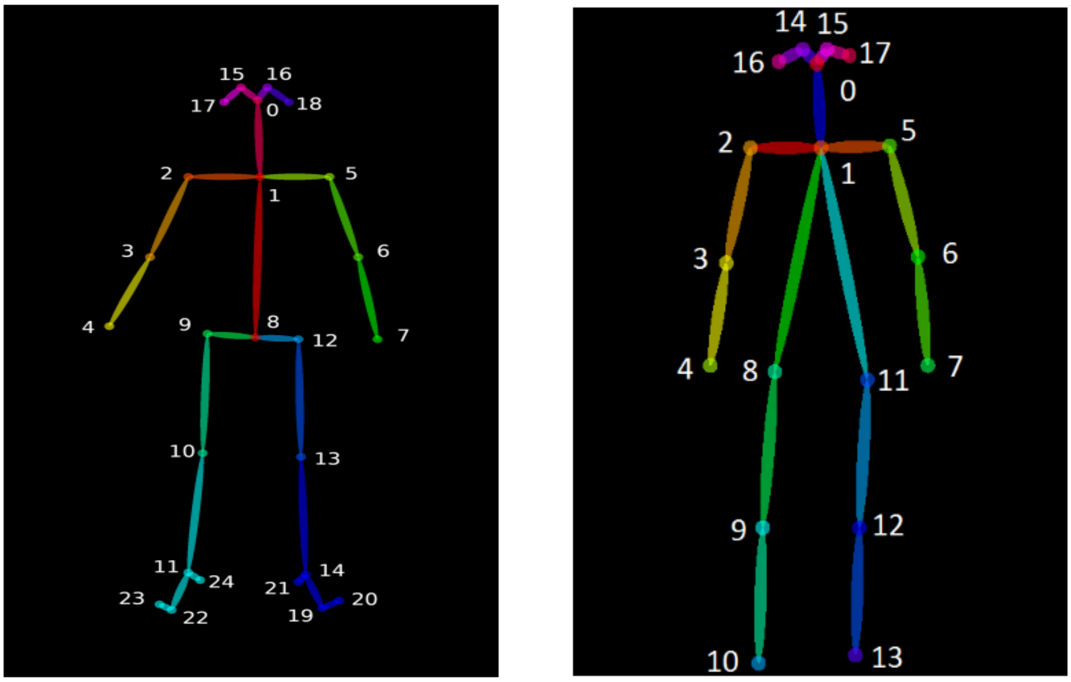
Figure 3.6 illustrates the overall pipeline of OpenPose. The system takes as input a color image (Fig. 3.6a). First, a feedforward network predicts simultaneously a set of 2D confidence maps S of body part positions (Fig. 3.6b) and a set of 2D vector fields L of part affinities, which encode the degree of association between body parts (Fig. 3.6c). The confidence maps and the affinity fields are then parsed by greedy inference (Fig. 3.6d) to produce as output the 2D locations of anatomical keypoints for each person in the image (Fig. 3.6e).

In this thesis, OpenPose was run on the experiments' videos of both cameras recording the experiment and the people pose data for each video frame was saved on a custom JSON file using the write_json flag. Each JSON file has a people array of objects, where each object has:

- an array pose_keypoints_2d containing the body part locations and detection confidence formatted as x1,y1,c1,x2,y2,c2,.... The coordinates x and y can be normalized to the range [0,1], [-1,1], [0, source size], [0, output size], etc., depending on the flag keypoint_scale, while c is the confidence score in the range [0,1]
- the arrays face_keypoints_2d, hand_left_keypoints_2d, and hand_right_keypoints_2d, analogous to pose_keypoints_2d
- the analogous 3-D arrays body_keypoints_3d, face_keypoints_3d, hand_left_keypoints_2d, and hand_right_keypoints_2d (if –3d is enabled, otherwise they will be empty). Their format is x1,y1,z1,c1,x2,y2,z2,c2,..., where c is simply 1 or 0 depending on whether the 3-D reconstruction was successful or not
- the body part candidates before being assembled into people (if –part_candidates is enabled).

Two possible pose formats are supported by OpenPose: The BODY_25 pose output format and the COCO pose output format. The keypoint ordering of these two body models is shown in figure 3.7.

The effectiveness of OpenPose is appreciable in figure 3.8, where two camera frames of two different test subjects during the experiment sessions are reported alongside the corresponding two outputs of OpenPose with the identified joints and limbs.



(a) BODY_25 Pose Output Format.

(b) COCO Pose Output Format.

Figure 3.7: The two possible pose output formats supported by OpenPose.

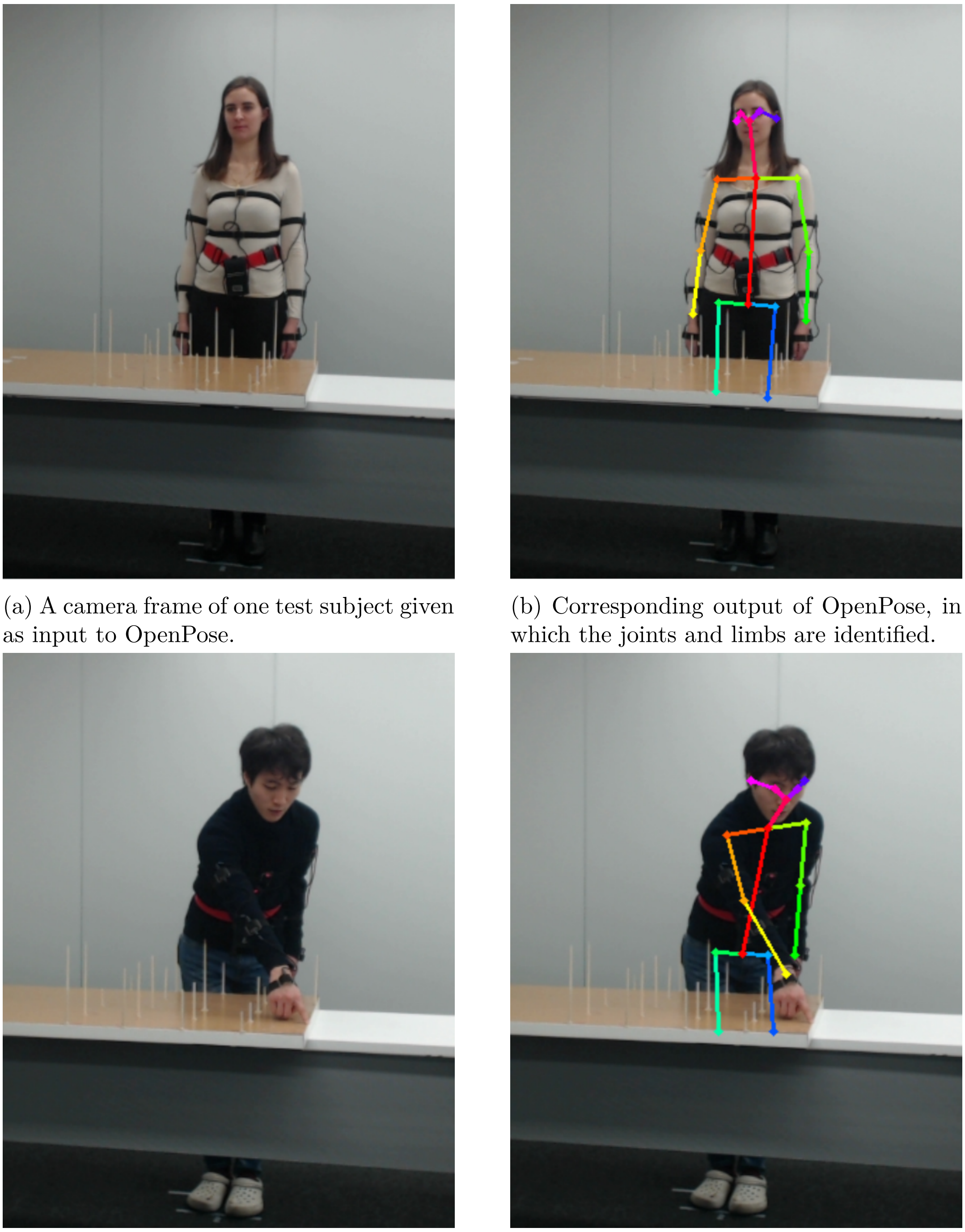
3.3.2 The reconstruction or triangulation problem

As hinted before, the reconstruction or triangulation problem consists in reconstructing three-dimentional geometric objects from matches obtained by stereo vision. More practically, the task is to determine the 3D coordinates of a point $\mathbf{P} = (x_w, y_w, z_w)$ in the world reference frame, given its two projections (u_l,v_l) and (u_r,v_r) on the two cameras' images. In order to solve this problem, it is crucial to determine the parameters of the camera projection function from 3D to 2D for the cameras involved. In the simplest case, this is represented by the camera matrices. Knowing the perspective projection matrices $\mathbf{M}_l$ and $\mathbf{M}_r$ of the two cameras (left and right) found during each single camera calibration, it is possible to write the following systems (equations 3.17 and 3.18):

$$s \cdot \begin{bmatrix} u_l \\ v_l \\ 1 \end{bmatrix} = \mathbf{M}_l \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3.17}$$

$$s \cdot \begin{bmatrix} u_r \\ v_r \\ 1 \end{bmatrix} = \mathbf{M}_r \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3.18}$$

where $(u_l,\ v_l)$, $(u_r,\ v_r)$ and $(x_w,\ y_w,\ z_w)$ are the coordinates of the same point in the left camera frame, right camera frame and in the world reference frame, respectively.



(a) A camera frame of one test subject given as input to OpenPose.

(b) Corresponding output of OpenPose, in which the joints and limbs are identified.

(c) A camera frame of one test subject given as input to OpenPose.

(d) Corresponding output of OpenPose, in which the joints and limbs are identified.

Figure 3.8: Two camera frames of two different subjects during the experiment sessions and the corresponding two outputs of OpenPose, with the identified joints and limbs.

The two previous systems give us the four following equations (3.19):

$$\begin{cases} u_l = \frac{m_{11_l} \cdot x_w + m_{12_l} \cdot y_w + m_{13_l} \cdot z_w + m_{14_l}}{m_{31_l} \cdot x_w + m_{32_l} \cdot y_w + m_{33_l} \cdot z_w + m_{34_l}} \\ v_l = \frac{m_{21_l} \cdot x_w + m_{22_l} \cdot y_w + m_{23_l} \cdot z_w + m_{24_l}}{m_{31_l} \cdot x_w + m_{32_l} \cdot y_w + m_{33_l} \cdot z_w + m_{34_l}} \\ u_r = \frac{m_{11_r} \cdot x_w + m_{12_r} \cdot y_w + m_{13_r} \cdot z_w + m_{14_r}}{m_{31_r} \cdot x_w + m_{32_r} \cdot y_w + m_{33_r} \cdot z_w + m_{34_r}} \\ v_r = \frac{m_{21_r} \cdot x_w + m_{22_r} \cdot y_w + m_{23_r} \cdot z_w + m_{24_r}}{m_{31_r} \cdot x_w + m_{32_r} \cdot y_w + m_{33_r} \cdot z_w + m_{34_r}} \end{cases} \tag{3.19}$$

After a few algebraic passages, the four equations become (equation 3.20):

$$\begin{cases} -m_{11_l} \cdot x_w - m_{12_l} \cdot y_w - m_{13_l} \cdot z_w + m_{31_l} \cdot x_w \cdot u_l + m_{32_l} \cdot y_w \cdot u_l + m_{33_l} \cdot z_w \cdot u_l = \\ \qquad = m_{14_l} - m_{34_l} \cdot u_l \\ -m_{21_l} \cdot x_w - m_{22_l} \cdot y_w - m_{23_l} \cdot z_w + m_{31_l} \cdot x_w \cdot v_l + m_{32_l} \cdot y_w \cdot v_l + m_{33_l} \cdot z_w \cdot v_l = \\ \qquad = m_{24_l} - m_{34_l} \cdot v_l \\ -m_{11_r} \cdot x_w - m_{12_r} \cdot y_w - m_{13_r} \cdot z_w + m_{31_r} \cdot x_w \cdot u_r + m_{32_r} \cdot y_w \cdot u_r + m_{33_r} \cdot z_w \cdot u_r = \\ \qquad = m_{14_r} - m_{34_r} \cdot u_r \\ -m_{21_r} \cdot x_w - m_{22_r} \cdot y_w - m_{23_r} \cdot z_w + m_{31_r} \cdot x_w \cdot v_r + m_{32_r} \cdot y_w \cdot v_r + m_{33_r} \cdot z_w \cdot v_r = \\ \qquad = m_{24_r} - m_{34_r} \cdot v_r \end{cases} \tag{3.20}$$

The system has 4 equations and 3 unknowns, so it is easily solvable using the Least Squares method with at least 1 point, by solving the following system (equation 3.21):

$$\mathbf{Q} \cdot \boldsymbol{b} = \boldsymbol{d} \tag{3.21}$$

where matrix $\mathbf{Q}$ and vector $\boldsymbol{d}$ contain all the known parameters (m_{11_l},..., m_{34_l}, m_{11_r},..., m_{34_r}) of the two perspective projection matrices $\mathbf{M}_l$ and $\mathbf{M}_l$ and the known image coordinates (u_l, v_l, u_r, v_r) of the considered point, while vector $\boldsymbol{b}$ contains the 3 unknown coordinates of the 3D point in the world frame (x_w, y_w, z_w).
The solution is obtained using the following equation (3.22):

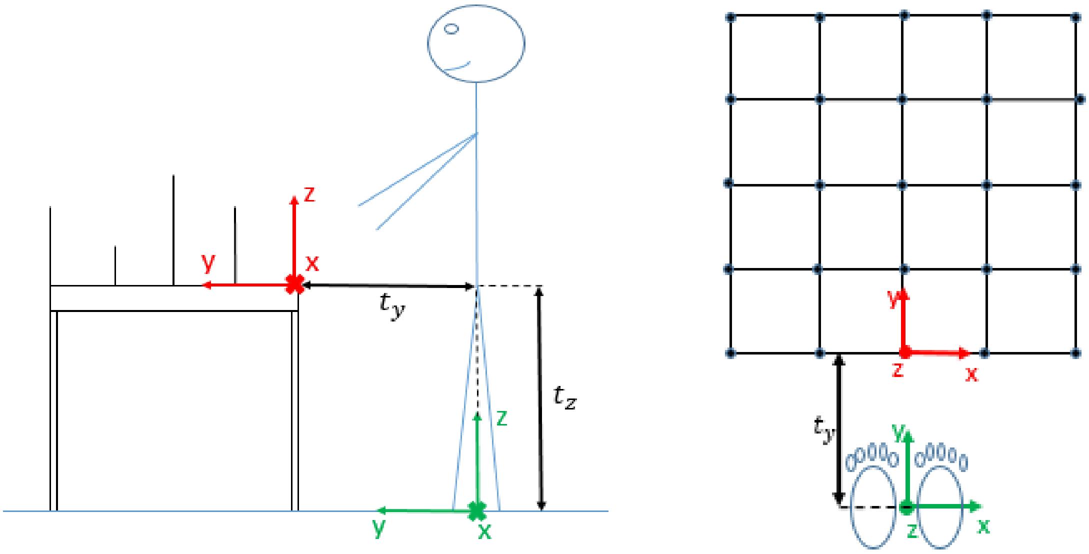
$$\boldsymbol{b} = (\mathbf{Q}^T \cdot \mathbf{Q})^{-1} \cdot \mathbf{Q}^T \cdot \boldsymbol{d} \tag{3.22}$$

if rank($\mathbf{Q}$)=3, which is assumed to be the case.

Once this system is solved, the 3D coordinates of the point that we want to reconstruct in the world frame are determined, thus solving the reconstruction problem.

3.4 Rototranslation from the camera world reference system to the global reference system

The last step is to report the just found 3D point **P** in the camera world reference frame to the global reference frame. Figure 3.9 shows the configuration of the two reference systems, seen from the side and from the top.



(a) Side view of the two reference frames. (b) Top view of the two reference frames.

Figure 3.9: Side view and top view of the two reference frames. In green, the global reference frame to which the positions determined by the IMU sensors are reported. In red, the camera world reference frame in which the positions determined by the two cameras are found.

In principle, the cameras calibration could be carried out directly on points which belong to the space defined by the global reference system. In this case, it was decided to calibrate the cameras on the three-dimensional grid, so it is necessary to bind the board to the global space. Obviously, the transformation between the two reference systems is known and it is expressed by a rototranslation matrix **RT**. The rotational part of such matrix is set to the identity matrix, since there is no rotation between the two reference systems (as shown in figure 3.9), while its translational part is the vector $\mathbf{t} = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T$ representing the difference between the position of a reference point seen from the camera world frame and the position of the same reference point seen from the global coordinate system. In this dissertation the neck point was selected as reference point. In order to report the three-dimensional point **P** in the camera world reference frame to the global reference frame, it is enough to simply pre-multiply such point in the camera frame written in homogeneous coordinates by the matrix **RT**. These steps are summarized in equations 3.23 and 3.24:

$$\mathbf{RT} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.23}$$

where $\boldsymbol{t} = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T = norm(neck_w - neck_g)$

$$\begin{bmatrix} x_g \\ y_g \\ z_g \\ 1 \end{bmatrix} = \mathbf{RT} \cdot \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3.24}$$

Chapter 4

The Kalman filter

The Kalman filter is a recursive state-space model based estimation algorithm named after Rudolf Emil Kalman, who in 1960 published his famous paper "A new approach to linear filtering and prediction problems" describing a recursive solution to the discrete-data linear filtering problem[41]. This method uses a dynamic model, measured control inputs and process measurements to estimate the process output. The estimation is composed of two distinct phases:

- **Time update** or **Prediction phase**: the state vector and the error covariance matrix are estimated, based on the system's mathematical model. Such an estimate is called *a-priori estimate* of the system.
- **Measurement update** or **Correction phase**: the a-priori state just estimated is corrected using an external measurement, in order to obtain a better estimate of the system state. This estimate is called *a-posteriori estimate* of the system.

The Kalman filter behaves then as a predictor-corrector algorithm, applying recursively the aforementioned procedure (figure 4.1).

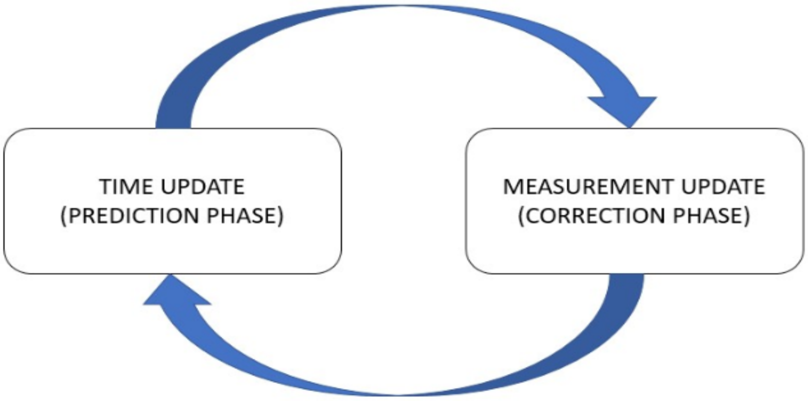


Figure 4.1: The recursive process of the Kalman filter.

4.1 The discrete Kalman filter

The content of this section is inspired by the work of Fabio Scibona [42], who used an extended formulation of the Kalman filter algorithm to combine measurements provided by three gyroscopes and a star sensor in a typical space scenario, for the development and validation of a navigation system for the attitude module based on low cost MEMS inertial sensors.

The discrete Kalman filter addresses the problem of estimating the state $x \in \Re^n$ of a discrete-time controlled process that is governed by the following state-transition equation at time k, called state model (equation 4.1):

$$\boldsymbol{x}_k = \boldsymbol{A}_{k-1} \cdot \boldsymbol{x}_{k-1} + \boldsymbol{B}_{k-1} \cdot \boldsymbol{u}_{k-1} + \boldsymbol{W}_{k-1} \cdot \boldsymbol{w}_{k-1} \tag{4.1}$$

and a measurement equation at time k, called measurement model (equation 4.2):

$$\boldsymbol{z}_k = \boldsymbol{H}_k \cdot \boldsymbol{x}_k + \boldsymbol{V}_k \cdot \boldsymbol{v}_k \tag{4.2}$$

where:

- $\boldsymbol{x}_k$ is the $n \times 1$ system state vector
- $\boldsymbol{u}_k$ is the $p \times 1$ system control input vector
- $\boldsymbol{A}_k$ is the $n \times n$ state-transition matrix, linking the system state $\boldsymbol{x}_{k-1}$ to the system state $\boldsymbol{x}_k$
- $\boldsymbol{B}_k$ is the $n \times p$ input matrix, linking the input $\boldsymbol{u}$ to the system state $\boldsymbol{x}_k$
- $\boldsymbol{W}_k$ is the $n \times n$ matrix linking the noise $\boldsymbol{w}$ to the system state $\boldsymbol{x}_k$
- $\boldsymbol{w}_k$ is the $n \times 1$ process noise vector, a Gaussian white noise with zero mean $E[\boldsymbol{w}_k] = 0$ and known covariance $E[\boldsymbol{w}_k \cdot \boldsymbol{w}_i^T] = \boldsymbol{Q}_k$ if i=k, zero otherwise
- $\boldsymbol{z}_k$ is the $m \times 1$ measurement vector
- $\boldsymbol{H}_k$ is the $m \times n$ observation matrix, linking the system state $\boldsymbol{x}_k$ to the measurement $\boldsymbol{z}_k$. It represents how the system state is registered by the sensors
- $\boldsymbol{V}_k$ is the $m \times m$ matrix linking the noise $\boldsymbol{v}$ to the measurement $\boldsymbol{z}_k$
- $\boldsymbol{v}_k$ is the $m \times 1$ measurement noise vector, a Gaussian white noise with zero mean $E[\boldsymbol{v}_k] = 0$ and known covariance $E[\boldsymbol{v}_k \cdot \boldsymbol{v}_i^T] = \boldsymbol{R}_k$ if i=k, zero otherwise.

If we assume that the noises $\boldsymbol{w}$ and $\boldsymbol{v}$ are uncorrelated, then the two covariance matrices $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$ are independent from each other.

Let us define the a-priori state vector at time k as $\hat{\boldsymbol{x}}_k^-$ and the a-posteriori state vector at time k as $\hat{\boldsymbol{x}}_k^+$. At each time step, the a-posteriori state is calculated as a linear combination of the a-priori state and the weighted difference between the actual measurement $\boldsymbol{z}_k$ and the estimated measurement $\boldsymbol{H}_k \cdot \hat{\boldsymbol{x}}_k^-$, as reported in formula 4.3:

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k \cdot (\boldsymbol{z}_k - \boldsymbol{H}_k \cdot \hat{\boldsymbol{x}}_k^-) \tag{4.3}$$

where:

- $(\boldsymbol{z}_k - \boldsymbol{H}_k \cdot \hat{\boldsymbol{x}}_k^-)$ is the term normally called residue
- $\boldsymbol{K}_k$ is the Kalman gain.

The Kalman gain allows to minimize the error covariance matrix $\boldsymbol{P}_k$ and can be calculated from formula 4.4:

$$\boldsymbol{K}_k = \frac{\boldsymbol{P}_k^- \cdot \boldsymbol{H}_k^T}{(\boldsymbol{H}_k \cdot \boldsymbol{P}_k^- \cdot \boldsymbol{H}_k^T + \boldsymbol{V}_k \cdot \boldsymbol{R}_k \cdot \boldsymbol{V}_k^T)} \tag{4.4}$$

From formulas 4.3 and 4.4, it is clear that $\boldsymbol{K}_k$ weights more the residue if the measurement covariance matrix $\boldsymbol{R}_k$ tends to zero:

$$\lim_{\boldsymbol{R}_k \to 0} \boldsymbol{K}_k = \boldsymbol{H}_k^{-1} \Longrightarrow \hat{\boldsymbol{x}}_k^+ \to \boldsymbol{H}_k^{-1} \cdot \boldsymbol{z}_k$$

Conversely, $\boldsymbol{K}_k$ weights less the residue if the a-priori state covariance matrix $\boldsymbol{P}_k^-$ tends to zero:

$$\lim_{\boldsymbol{P}_k^- \to 0} \boldsymbol{K}_k = 0 \Longrightarrow \hat{\boldsymbol{x}}_k^+ \to \hat{\boldsymbol{x}}_k^-$$

The Kalman gain allows therefore to optimally estimate the system state, weighting the measurement proportionally to its reliability. Its reliability is defined by the comparison between the covariance of the measurement expected from the mathematical model $\boldsymbol{P}_k^-$ and the covariance of the acquired measurement $\boldsymbol{R}_k$.

Once the a-posteriori state has been calculated from formula 4.3, the state covariance matrix is updated using formula 4.5 in order to find the a-posteriori state covariance matrix:

$$\boldsymbol{P}_k^+ = (\boldsymbol{I} - \boldsymbol{K}_k \cdot \boldsymbol{H}_k) \cdot \boldsymbol{P}_k^- \tag{4.5}$$

This concludes the correction phase of the Kalman filter.

In the prediction phase, the new a-priori state is calculated based on the mathematical model of the system, according to the state equation 4.6:

$$\hat{\boldsymbol{x}}_{k+1}^- = \boldsymbol{A}_k \cdot \hat{\boldsymbol{x}}_k^+ + \boldsymbol{B}_k \cdot \boldsymbol{u}_k \tag{4.6}$$

Then, the new covariance matrix is calculated using formula 4.7:

$$\boldsymbol{P}_{k+1}^- = \boldsymbol{A}_k \cdot \boldsymbol{P}_k^+ \cdot \boldsymbol{A}_k^T + \boldsymbol{W}_k \cdot \boldsymbol{Q}_k \cdot \boldsymbol{W}_k^T \tag{4.7}$$

The discrete Kalman filter algorithm is described in figure 4.2.

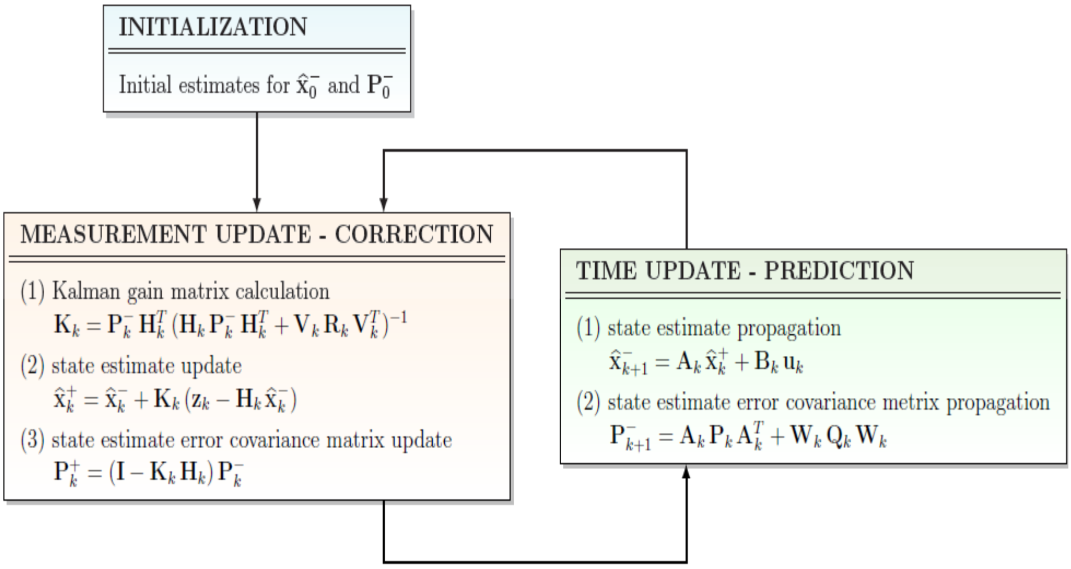

INITIALIZATION
Initial estimates for $\hat{x}_0^-$ and P_0^-
MEASUREMENT UPDATE - CORRECTION
(1) Kalman gain matrix calculation
$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1}$
(2) state estimate update
$\hat{x}_k^+ = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-)$
(3) state estimate error covariance matrix update
$P_k^+ = (I - K_k H_k) P_k^-$
TIME UPDATE - PREDICTION
(1) state estimate propagation
$\hat{x}_{k+1}^- = A_k \hat{x}_k^+ + B_k u_k$
(2) state estimate error covariance metrix propagation
$P_{k+1}^- = A_k P_k A_k^T + W_k Q_k W_k$

Figure 4.2: The discrete Kalman filter algorithm (from [42]).

4.2 The Kalman filter fusion algorithm

In this work, the discrete Kalman filter is used to integrate the positions of the human joints in the global reference frame obtained from the inertial sensors with the ones reconstructed from the the cameras.
Each state vector $\mathbf{x}$ is composed of the (x,y,z) coordinates of the global position of one joint of the user in the environment. $\mathbf{A}$ is a 3×3 identity matrix, in order to incorporate directly the IMU measurements, and $\mathbf{B}$ is a null matrix since there are no control inputs. The noise matrices $\mathbf{W}$ and $\mathbf{V}$ are 3×3 identity matrices, as well as the observation matrix $\mathbf{H}$. The process noise covariance matrix $\mathbf{Q}$ is a diagonal matrix because state vector variables are not correlated. Its diagonal terms correspond to the mean error of the inertial measurements. The measurement noise covariance matrix $\mathbf{R}$ is a diagonal matrix because measurement vector variables are not correlated. Its diagonal terms correspond to the mean error of the camera measurements.

The Kalman filter based fusion algorithm used in this thesis works according to the flowchart reported in figure 4.3.
Before starting the main loop, the initial a-priori state vectors $\hat{\boldsymbol{x}}_0^-$ (one for each joint being tracked) are estimated as the global positions found from the first frame of the two cameras, when the human is in the rest position. Also the initial a-priori covariance matrix $\boldsymbol{P}_0^-$ is initialized as matrix $\mathbf{Q}$.

When a new measurement arrives at time k, the first step to be executed is the correction phase of the Kalman filter algorithm:

1. first, the Kalman gain at time k $\boldsymbol{K}_k$ is calculated according to formula 4.4

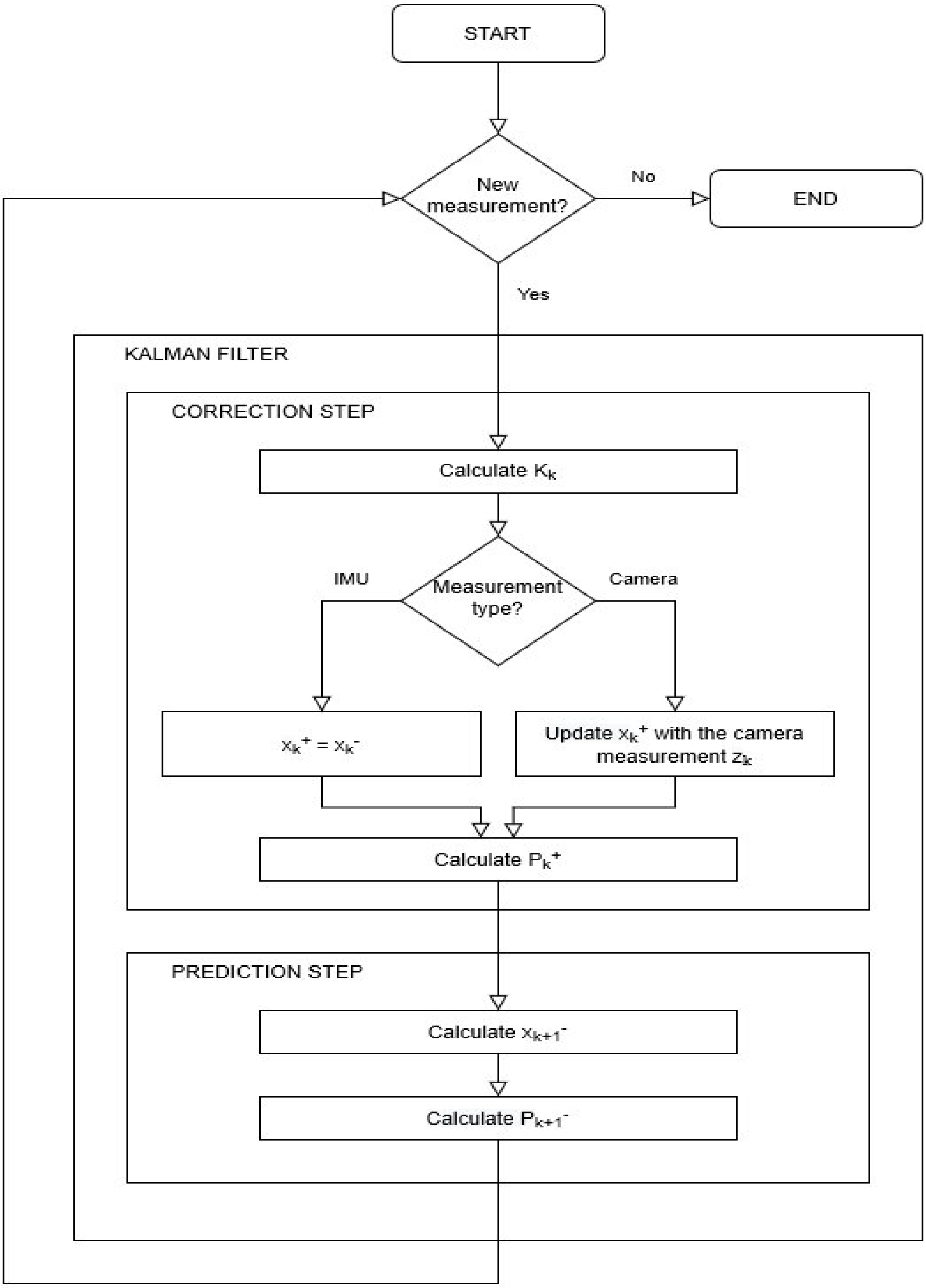


Figure 4.3: The fusion algorithm diagram.

2. then, the a-posteriori state vector at time k $\hat{\boldsymbol{x}}_k^+$ is computed:
    - if the measurement comes from an IMU sensor, the a-posteriori state vector coincides with the a-priori state vector found from the IMU sensor in the prediction step performed at time *k-1*
    - if the measurement comes from the cameras ($\boldsymbol{z}_k$), the a-posteriori state vector is updated using the camera measurement in order to eliminate the error accumulation in the previous a-priori estimate and thus compute an improved a-posteriori estimate of the global position $\hat{\boldsymbol{x}}_k^+$, as reported in formula 4.3
3. lastly, the a-posteriori error covariance matrix at time k $\boldsymbol{P}_k^+$ is updated according to formula 4.5.

Afterwards, the prediction phase is executed:

1. first, a new a-priori state vector estimate of the global position of the user's considered joint at time *k+1* $\hat{\boldsymbol{x}}_{k+1}^-$ is computed by incorporating the position measurement of the corresponding IMU sensor, following formula 4.6. This estimate will be used as starting point in the next cycle to compute the a-posteriori state vector at time *k+1*
2. finally, a new a-priori estimate of the error covariance matrix at time *k+1* $\boldsymbol{P}_{k+1}^-$ is calculated from formula 4.7. Also this estimate will be used in the next cycle to compute both the a-posteriori error covariance matrix and the Kalman gain at time *k+1*.

The loop execution continues as long as new measurements are received.

Thereby, the prediction step will be executed with the inertial sensors rate (*200 Hz*) and the correction step will be executed with the cameras rate (*15 Hz*).

Chapter 5

Experiments

5.1 Experiments Instrumentation

5.1.1 Board

The grid used for the purpose of the experiments was built using a flat wooden surface whose dimensions were $60cm \times 60cm$. Several wooden sticks of different heights were fixed on top of the board and any two sticks were placed 15 cm apart, both along the length and width directions of the grid. The sticks are either 0 cm, 5 cm, 10 cm, 15 cm or 20 cm tall. A picture of the board used for the experiments is shown in figure 5.1.

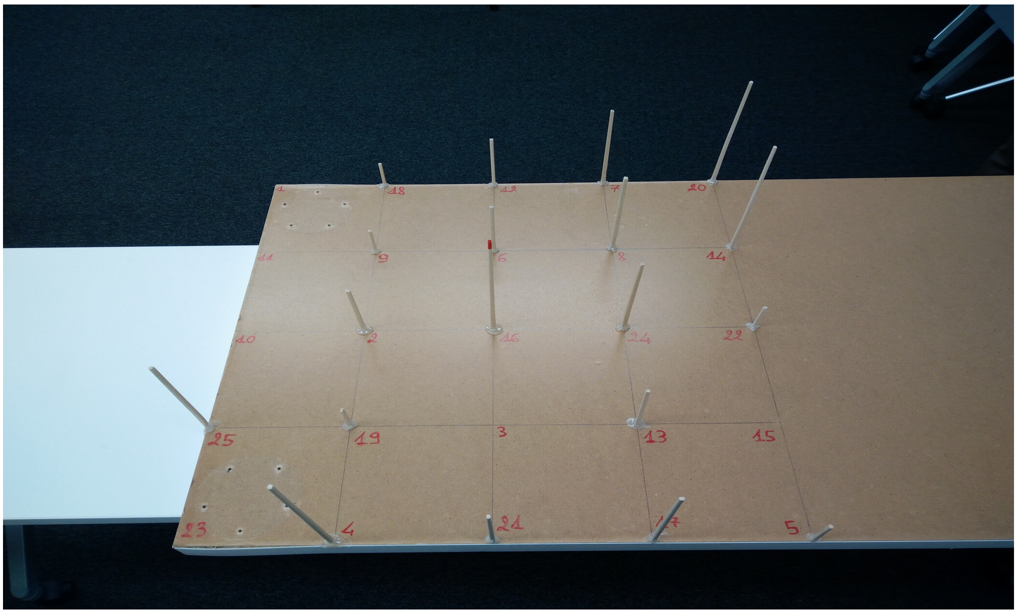


Figure 5.1: The grid used in the experiments.

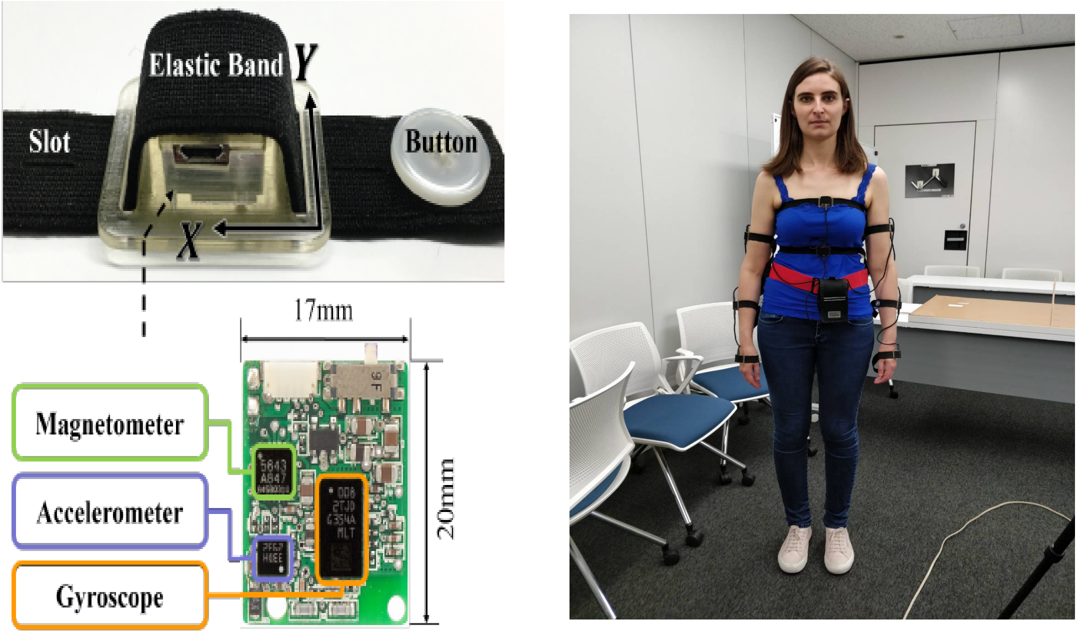
5.1.2 IMU sensors

The inertial sensors used for the experiments are the WB-4R (Waseda Bioinstrumentation 4R) Inertial Measurement Units, shown in figure 5.2a. Each of these sensors contains a tri-axial accelerometer, a tri-axial gyroscope and a tri-axial magnetometer. The specifications of these sensors are reported in Table 5.1. The sampling frequency of the WB-4R is *200 Hz*. More detailed information about these sensors can be found in [35].

	Accelerometer	Gyroscope	Magnetometer
	(LIS331DLH)	(LYPR540AH)	(HMC5843)
Axis	3-axis	3-axis	3-axis
Range	±2/ ± 4/ ± 8 [G]	±400/ ± 1600 [deg/s]	±4 [Gauss]
Resolution	1/2/3.9 [mG/digit]	3.2/0.8 [mV/dps]	12 [bit]
Bandwidth	25/50/500 [Hz]	140 [Hz]	50 [Hz]

Table 5.1: Specifications of WB-4R IMU sensors.

Eight IMU sensors are placed on the human body: one on the right upper arm, one on the left upper arm, one on the right forearm, one on the left forearm, one on the right hand, one on the left hand, one on the torso and one right under the neck. The sensors are fixed on the body with elastic bands, as shown in figure 5.2b, which makes it easy to adjust them to different body parts and to different partecipants.



(a) WB-4R Inertial Measurement Unit (IMU).

(b) IMU sensors placement on the body

Figure 5.2: WB-4R IMU sensor and IMU placement on the body.

The eight IMU sensors are connected to a central board with micro USB cables, in a daisy chain fashion, and they communicate via CAN bus. The data transmission between the central board and the computer is implemented using Bluetooth 2.1 (Class 1), ensuring the synchronization between sensors and allowing a large workspace for the partecipants.

5.1.2.1 Inertial sensors calibration

The first operation to be performed when dealing with inertial measurement units is their calibration, an essential step in order to set standards for the IMU attitude and reduce errors caused by inaccurate sensor measurements.

Accelerometer calibration In order to calibrate the accelerometer, the following tools are needed and they are shown in figure 5.3:

- calibration box: it is an empty box with flat and perpendicular surfaces and two cuts on the case to pass the USB cable
- WB-4R IMU sensor to be calibrated and central board
- horizontal plate: it can be made from a flat plate and three bolts, three spring washers and six nuts
- thin double side tape to fix the sensor to the calibration box
- level gauge

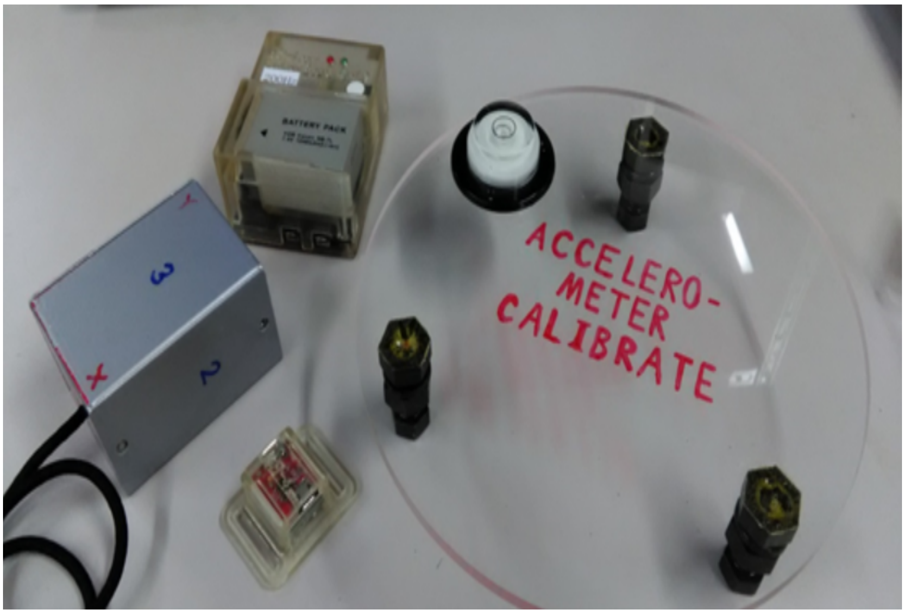
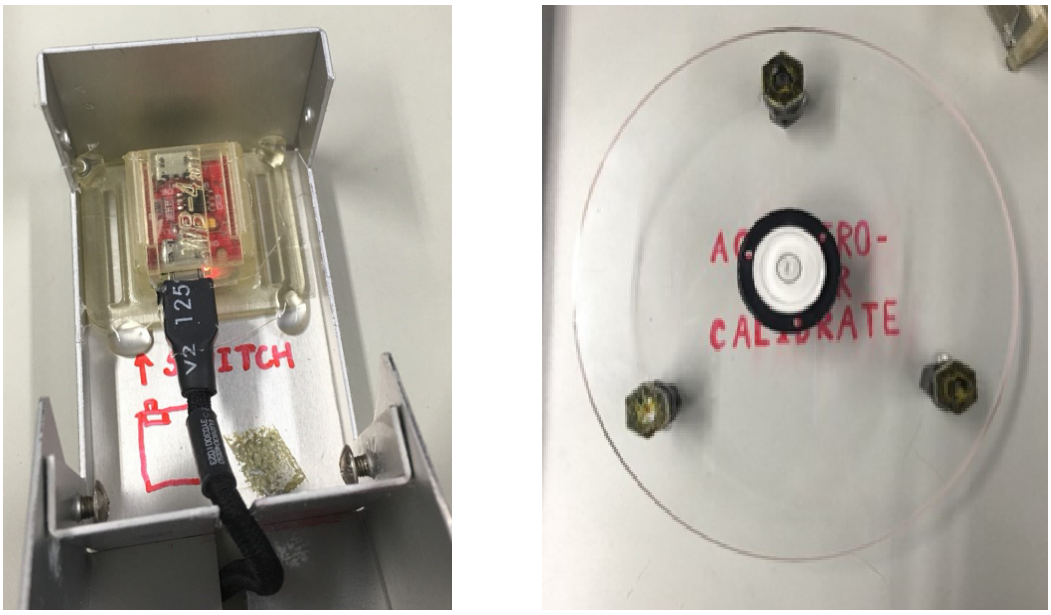


Figure 5.3: Equipment necessary for the accelerometer calibration.

First of all, the IMU sensor needs to be fixed to the calibration box in the correct direction drawn on the inside of the calibration box itself using double side tape. This positioning is shown in figure 5.4a. Then, the height of the three legs of the horizontal plate needs to be adjusted to make it horizontal, using the level gauge to check the correct inclination. This step is shown in figure 5.4b.



(a) Correct positioning of the IMU sensor inside the calibration box.

(b) Horizontal plate with the level gauge to check its inclination.

Figure 5.4: Correct positioning of theIMU sensor inside the calibration box and horizontal inclination checking using the level gauge.

The IMU sensor needs to be connected to the central board via Bluetooth and then the calibration box needs to be positioned on the horizontal plate with one face down and kept still for 10 seconds, the time needed by the software to take 100 samples. Once the 100 samples are taken, the calibration box can be rotated onto another face and the same procedure is repeated. After taking samples from all six faces, the calibration of the accelerometer is complete. The calibration box positioned on the horizontal plate during the accelerometer calibration is shown in figure 5.5.

Gyroscope calibration In order to calibrate the gyroscope, the following tools are needed:

- calibration box
- WB-4R IMU sensor to be calibrated and central board
- turn table

First of all, the sensor needs to be fixed with double side tape inside the calibration box and connected to the central board via Bluetooth. The calibration box and the central board need to be placed on the turn table and the rotation speed of the turn table needs to be set

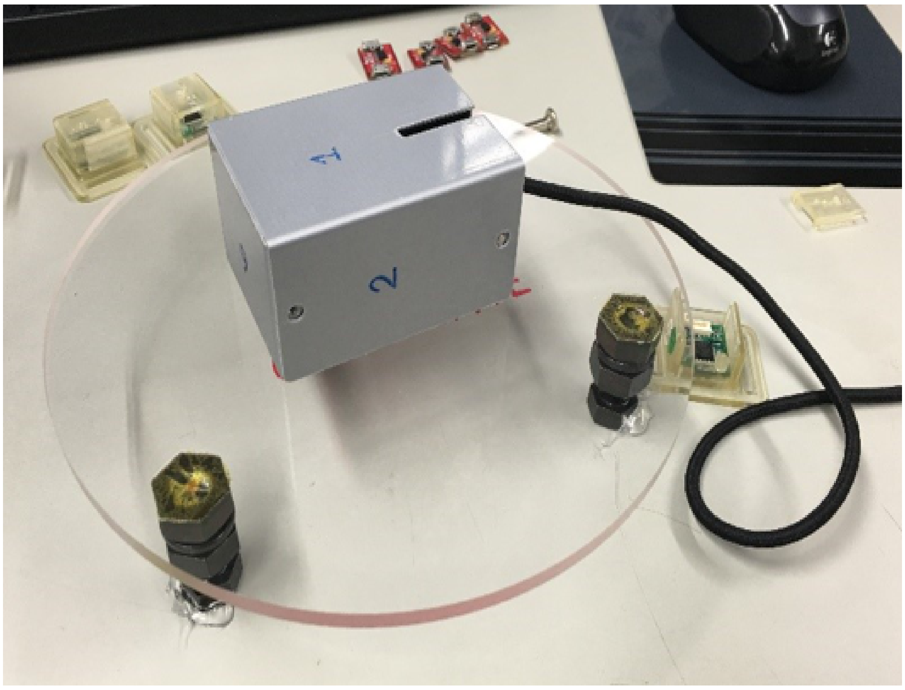


Figure 5.5: IMU accelerometer calibration.

at 45 rounds per minute. At first, the sensor is kept static for a few seconds with one face of the calibration box placed down on the turn table. Then, the rotation of the turn table is activated to conclude the calibration of that axis. The same procedure is repeated for every face of the calibration box, until the gyroscope calibration is finished. The calibration box and the central board positioned on the turn table during the gyroscope calibration are shown in figure 5.6.

5.1.3 Cameras

The cameras used for the experiments are two Logitech HD Pro Webcams C920, shown in figure 5.7. They are able to deliver full HD video (1080p at 30fps) and clear, stereo sound. The 78-degree field of view can frame up to two people at once and two integrated microphones capture audio from every angle. They are compatible with Windows 10 or later, Windows 8 and Windows 7 and they work in USB Video Device Class (UVC) mode with different supported video-calling clients. The technical specifications of such cameras are reported in table 5.2.

5.2 Experiments Setup

The experiments were performed at prof. Takanishi Laboratory of Waseda University, located at TWIns, in Tokyo, Japan. The grid was positioned on top of a desk, at a height of approximately 75 cm. The test subject was positioned on one side of the board. The two cameras were positioned on the other side, at a measured distance of approximately 213

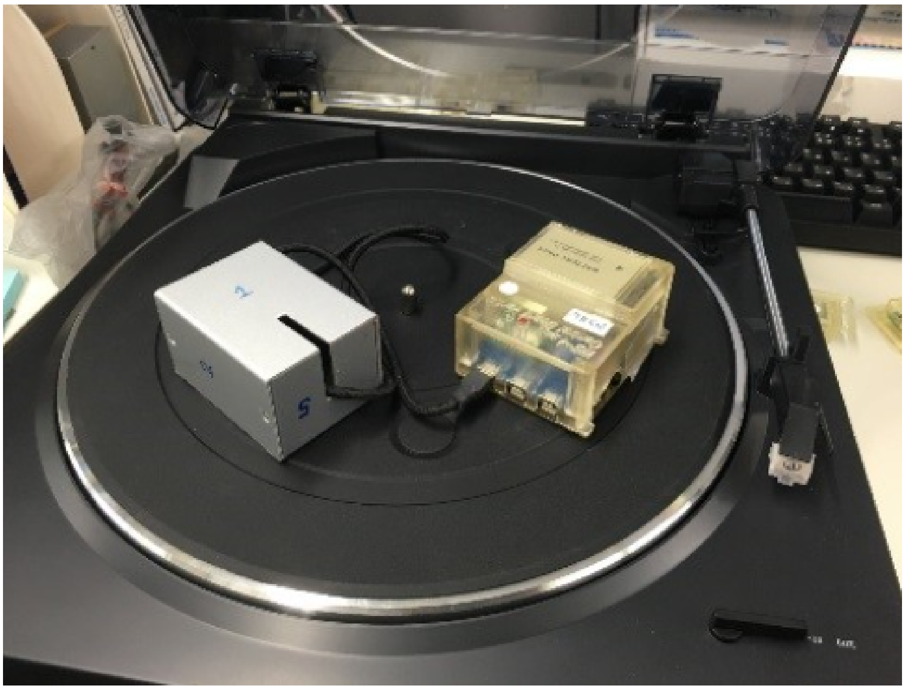


Figure 5.6: IMU gyroscope calibration.

Figure 5.7: Logitech HD Pro Webcams C920.

cm from the board, and were mounted on top of two tripods at a height of approximately 117.5 cm. The measured distance between the two camera centers was approximately 99 cm. The room setup can be seen in figure 5.8.

Two experiments were designed, in which the test subject had to perform a series of movements on the grid while the inertial sensors were placed on his body and the two cameras were recording his motion. The IMU sensors recorded the accelerations and the

	Logitech HD Pro Webcam C920
Dimensions (Height x Width x Depth)	Without clip: 29 mm x 94 mm x 24 mm Including clip: 43.3 mm x 94 mm x 71 mm
Cable Length	1.5 m
Max Resolution	1080p/30fps - 720p/30fps
Focus type	autofocus
Built-in mic	stereo

Table 5.2: Specifications of Logitech HD Pro Webcams C920.

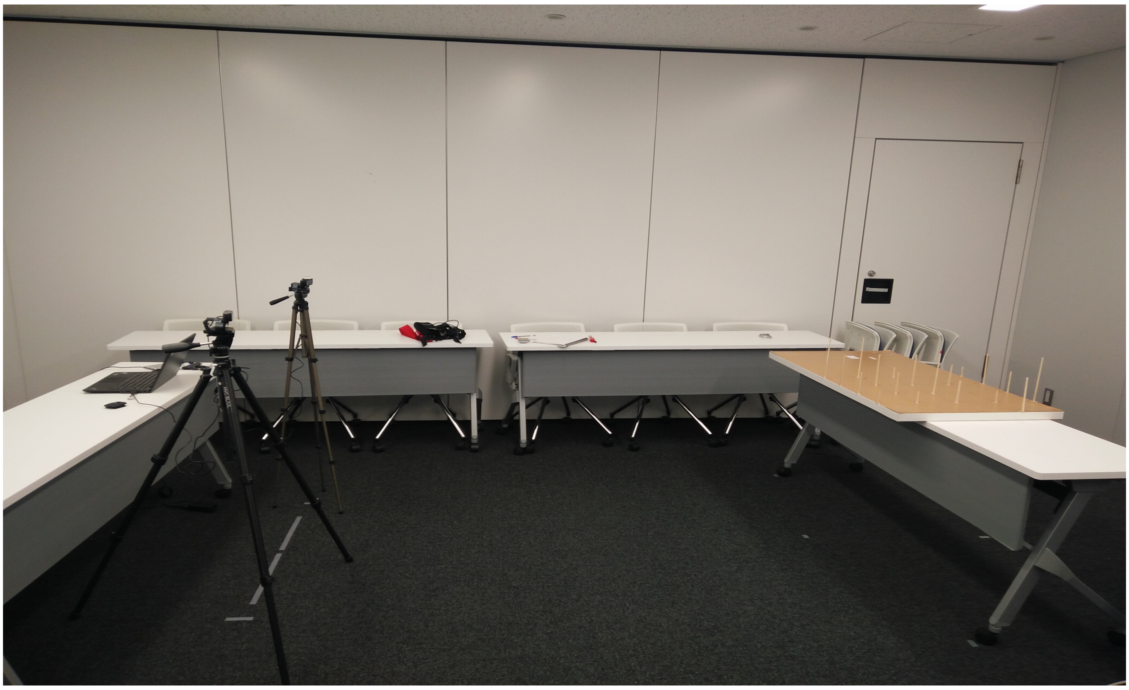
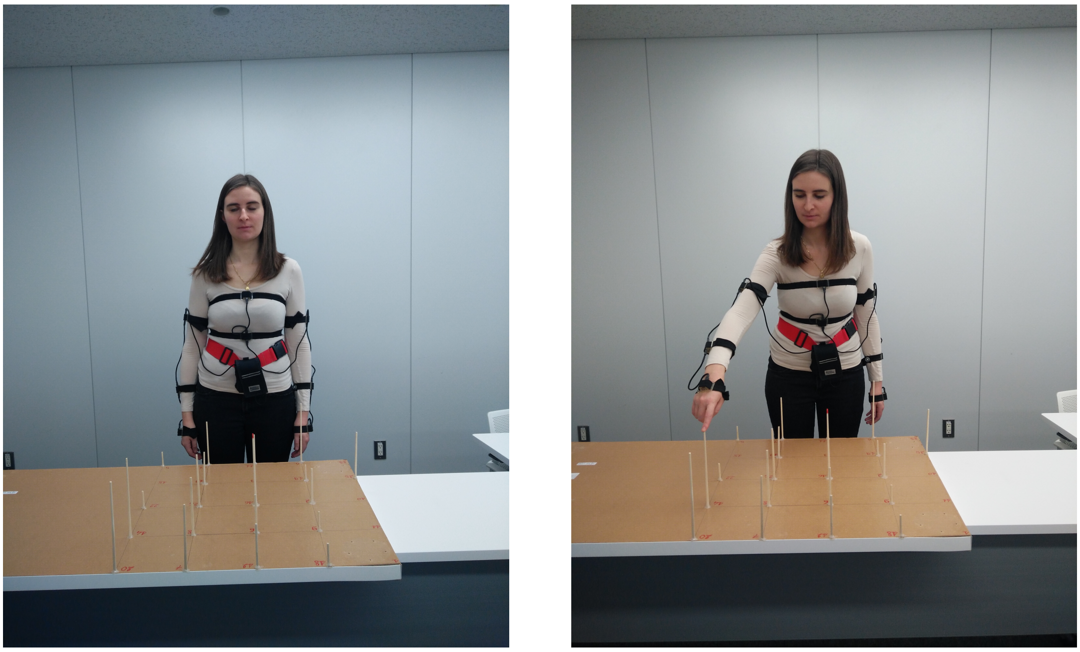


Figure 5.8: Room setup for the experiments.

angular velocities along the three axes (x,y,z) at a rate of *200 Hz*, so once every 5 ms, while the two cameras recorded the subject's motion at a rate of *15 fps*. For both experiments, the subject started from the rest position, which consists in standing straight with both arms lowered along the sides. The rest position, sometimes also called neutral condition or N-pose, is shown in figure 5.9a.

- The first experiment was of short duration in time and consisted in the test subject touching with his right index finger the tips of the 4 wooden sticks fixed on the 4 corners of the three-dimensional grid. After every touch, the subject returned to the rest position for a couple of seconds.

- The second experiment was of longer duration in time and consisted in the test subject touching with his right index finger the tips of all 25 wooden sticks, following a predefined sequence. This time, the subject did not return to the rest position until the end of the experiment. One of the test subjects performing the second experiment is shown in figure 5.9b.



(a) One of the test subjects standing in the rest position, also called neutral condition or N-pose.

(b) One of the test subjects while performing the second experiment, which is the longer one.

Figure 5.9: Experiments setup.

Four young subjects were recruited from the WB and Musical groups of prof. Takanishi Laboratory at Waseda University and participated in the experiments in two different sessions held in two different days. The subjects were chosen for having variety on height, weight and sex, as reported in table 5.3. The experiments were performed with different subjects to make sure that the results were not biased by the particular position of the sensors on a single subject or by his motion during the test. Each subject repeated each experiment three times, at different speeds.

Subject	S1	S2	S3	S4	Mean
Age	23	27	25	20	23.75
Height [cm]	175	162	180	173	172.5
Weight[kg]	60	68	80	75	70.75

Table 5.3: Anthropometric information of experimental subjects.

Chapter 6

Results

Using the data collected from the IMU sensors and from the cameras and following the procedure explained in the previous chapters, it is possible to reconstruct the human upper body segments.

6.1 Calibration movements

First of all, the calibration movements of the arms along the sagittal and coronal planes were reconstructed. The results of the calibration movements reconstructions are only qualitative and they are presented to show the behaviour of the system in simple cases. In the following body plots, only the subject's shoulders and arms are displayed.
Figure 6.1 shows the three-dimensional reconstruction of the calibration movement of the right arm along the sagittal plane. The test subject with the arm raised forward at 90 degrees and the corresponding reconstruction are reported in figures 6.2a and 6.2b.
Figure 6.3 shows instead the three-dimensional reconstruction of the calibration movement of the right arm along the coronal plane. The test subject with the arm raised sideways at 90 degrees and the corresponding reconstruction are reported in figures 6.4a and 6.4b, respectively.

6.2 Complex movements

When considering more complex movements, it is possible to compare three approaches:

1. the first considered approach consists in reconstructing the human upper body using only the measurements provided by the inertial sensors, completely neglecting the cameras data
2. the second method performs the reconstruction based only on the cameras measurements, completely neglecting the IMU sensors data
3. the third approach takes into account both the inertial and the cameras data, integrating them with a Kalman filter fusion algorithm with the aim of improving the final skeleton reconstruction.

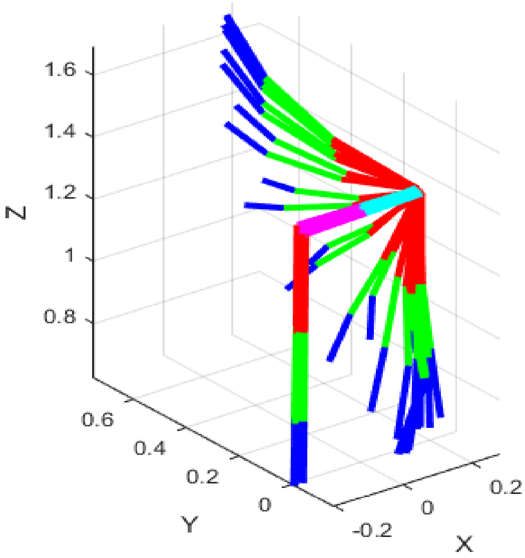
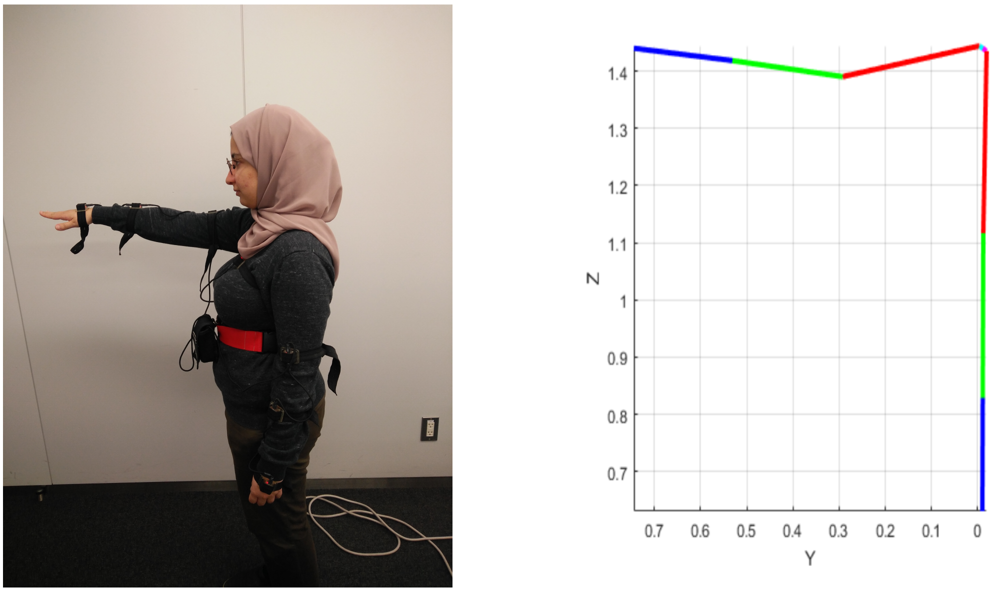


Figure 6.1: Three-dimensional reconstruction of the first calibration movement. The right arm is rotated forward, along the sagittal plane. Its position in the global reference frame is plotted once every 20 time steps.



(a) Test subject with the arm raised forward at 90 degrees.

(b) Reconstruction of the test subject with the arm raised forward at 90 degrees.

Figure 6.2: Test subject with the arm raised forward at 90 degrees and corresponding reconstruction.

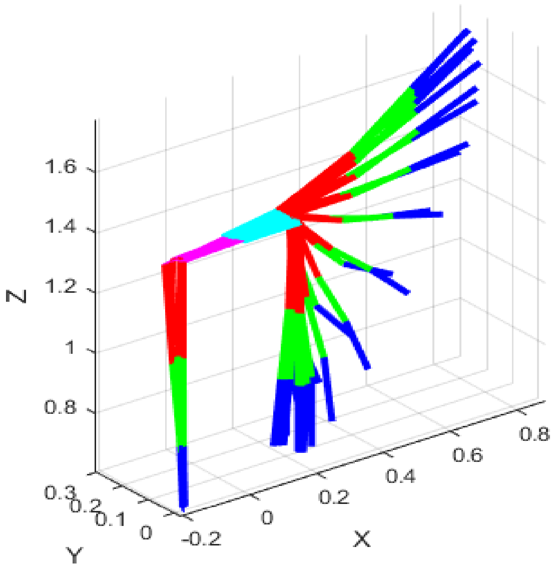
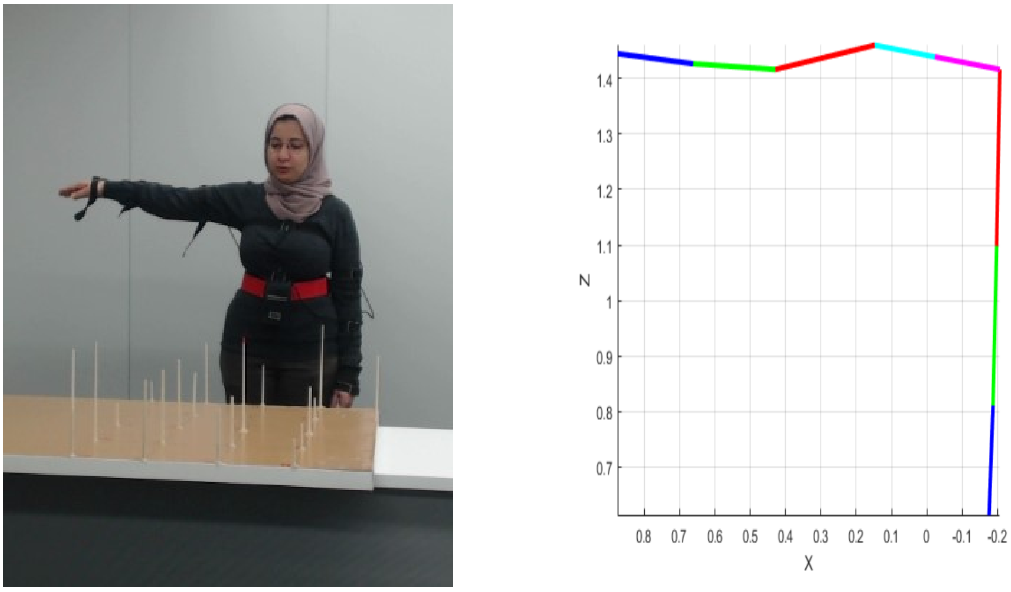


Figure 6.3: Three-dimensional reconstruction of the second calibration movement. The right arm is rotated sideways, along the coronal plane. Its position in the global reference frame is plotted once every 20 time steps.



(a) Test subject with the arm raised sideways at 90 degrees.

(b) Reconstruction of the test subject with the arm raised sideways at 90 degrees.

Figure 6.4: Test subject with the arm raised sideways at 90 degrees and corresponding reconstruction.

In order to validate the experiments, the right index finger positions in the global reference frame at the time instants in which the subject touched the tip of each stick are compared to the real positions of the sticks' tips. To do so, a minimization procedure is performed applying an algorithm called Iterative Closest Point (ICP). The ICP algorithm was first introduced by Paul J. Besl and Neil D. McKay in their 1992 paper "A method for registration of 3-D shapes" [43]. It is widely employed to minimize the difference between two clouds of points, in order to align three-dimensional shapes for object recognition and reconstruct 2D or 3D surfaces from different scans. In the Iterative Closest Point algorithm, one reference point cloud is kept fixed and the other one is transformed to find the best match with the reference one. Every point in one data set is coupled with the closest point in the other data set, forming correspondence pairs. Then the algorithm iteratively refines the transformation between the two point clouds, which is a combination of translation and rotation, in order to minimize a point-to-point error metric, usually the sum of the squared differences between the coordinates of the matched pairs. The process is iterated until the error becomes smaller than a threshold or it stabilizes.
The mathematical formulation of the ICP algorithm is the following: given two corresponding point sets $X = x_1, ..., x_{N_x}$ and $P = p_1, ..., p_{N_p}$, the goal is to find a rotation $\mathbf{R}_{ICP}$ and a translation $\mathbf{t}_{ICP}$ that minimize the sum of the squared errors between each matching pair of the two point clouds:

$$E_{ICP} = \frac{1}{N_p} \cdot \sum_{i=1}^{N_p} ||x_i - \mathbf{R}_{ICP} \cdot p_i - \mathbf{t}_{ICP}||^2$$

where x_i and p_i are corresponding points.
If the correct correspondences are known, it is possible to calculate the correct relative rotation and translation in closed form.

The reported ICP errors are calculated from experiments repeated by different subjects. The shown temporal plots instead are relative to some subjects only, since all the test subjects presented similar trends over time.

6.2.1 IMUs-only reconstruction

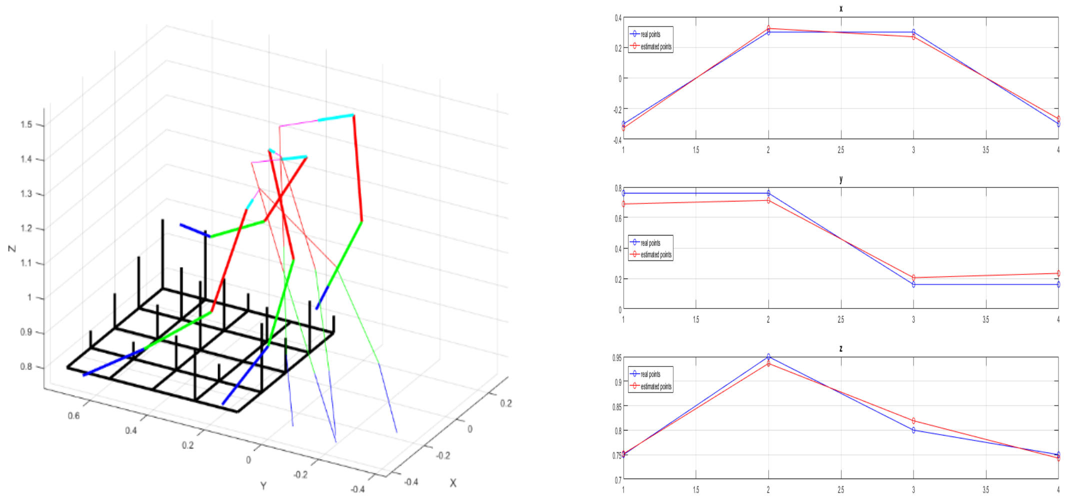
The first implementation involves the reconstruction of the human upper body using only the measurements provided by the inertial sensors. The detailed procedure is explained in chapter 2.

6.2.1.1 Short movement

Figure 6.5a shows the three-dimensional reconstruction of the human shoulders and arms at the 4 moments in which the test subject touches the grid's 4 corners. The thicker arm is the right one, which moves to touch the tops of the grid's sticks.
The calculated ICP error E_{ICP} between the estimated and the real point clouds obtained after the ICP minimization procedure is:

$$E_{ICP} = 6.72\ cm$$

The real x,y,z coordinates of each of the 4 corners in the global reference frame (in blue) and the estimated x,y,z coordinates of each of the 4 corners in the global reference frame obtained after the minimization procedure (in red) are compared in figure 6.5b.



(a) 3D reconstruction of the human upper body at the 4 moments in which the test subject touches the grid's 4 corners using inertial sensors only.

(b) Difference between the 4 corners real positions (in blue) and their estimated positions using inertial sensors only along the x, y and z directions (in red).

Figure 6.5: Three-dimensional reconstruction of the test subject touching the 4 corners of the grid and difference between the 4 corners real and estimated positions in the global reference frame, using inertial measurements only.

6.2.1.2 Long movement

The long movement consisted in touching sequentially the tips of all 25 sticks fixed on the board. The calculated ICP error E_{ICP} between the estimated and the real point clouds obtained after the ICP minimization procedure is:

$$E_{ICP} = 22.57\ cm$$

Figure 6.6 shows the difference between the real x,y,z coordinates of each stick's tip in the global reference frame (in blue) and the estimated x,y,z coordinates of each stick's tip in the global reference frame obtained after the ICP minimization procedure (in red).
In figure 6.7, instead, the error evolution in time is displayed. On the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis is plotted the norm of the error vector. The error at each point is the difference between the real position of the tip of the stick in the global reference frame and the estimated position of the tip of the stick in the global reference frame.

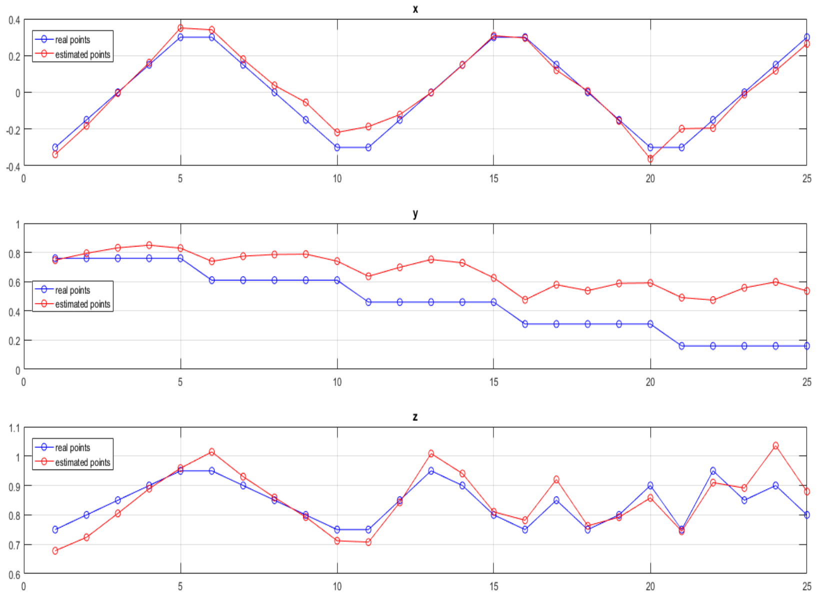


Figure 6.6: Difference between the 25 points real positions (in blue) and their estimated positions using inertial sensors only along the x, y and z directions (in red).

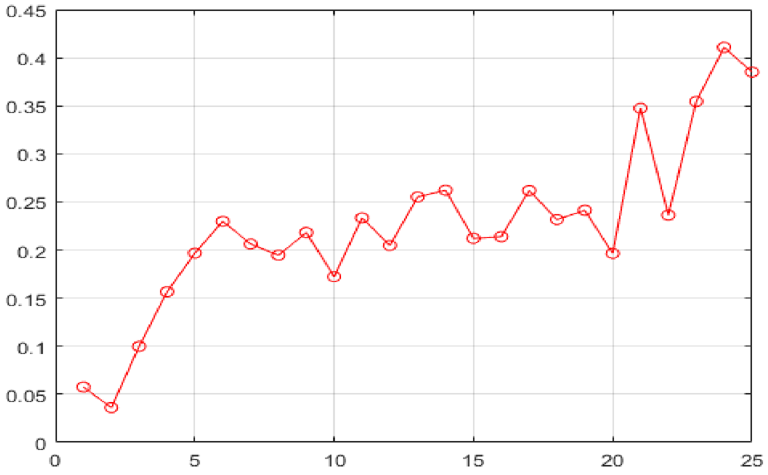


Figure 6.7: The error evolution in time using inertial sensors only. On the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis the norm of the error vector is plotted.

6.2.2 Vision-only reconstruction

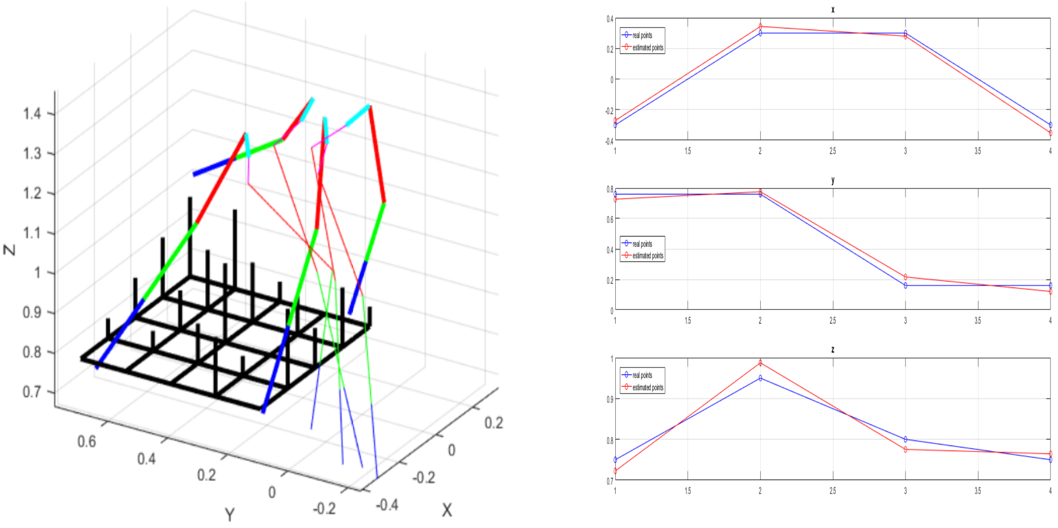
The second method treated in this dissertation features a pure vision reconstruction. The procedure is explained in chapter 3.

6.2.2.1 Short movement

The three-dimensional reconstruction of the human shoulders and arms at the 4 moments in which the test subject touches the tip of the 4 sticks fixed at the grid's 4 corners is shown in figure 6.8a. As explained before, the thicker arm is the right one, which the test subject moves to touch the tips of the sticks.
In this case, the calculated ICP error between the estimated and the real point clouds obtained after the ICP minimization procedure is:

$$E_{ICP} = 6.05\ cm$$

Figure 6.8b compares the real x,y,z coordinates of each of the 4 corners in the global reference frame (in blue) and the estimated x,y,z coordinates of each of the 4 corners in the global reference frame obtained after the minimization procedure (in red).



(a) 3D reconstruction of the human upper body at the 4 moments in which the test subject touches the 4 corners of the grid using cameras only.

(b) Difference between the 4 corners real positions (in blue) and their estimated positions using cameras only along the x, y and z directions (in red).

Figure 6.8: Three-dimensional reconstruction of the test subject touching the 4 corners of the grid and difference between the 4 corners real and estimated positions in the global reference frame, using cameras measurements only.

6.2.2.2 Long movement

For the 25 points experiment, the calculated ICP error between the estimated and the real point clouds obtained after the ICP minimization procedure using vision measurements only is:

$$E_{ICP} = 12.08\ cm$$

Figure 6.9 shows the difference between the real x,y,z coordinates of each stick's tip in the global reference frame (in blue) and the estimated x,y,z coordinates of each stick's tip obtained after the ICP minimization procedure (in red).
The error evolution in time in the cameras-only method is displayed in figure 6.10. As before, on the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis the norm of the error vector is plotted.

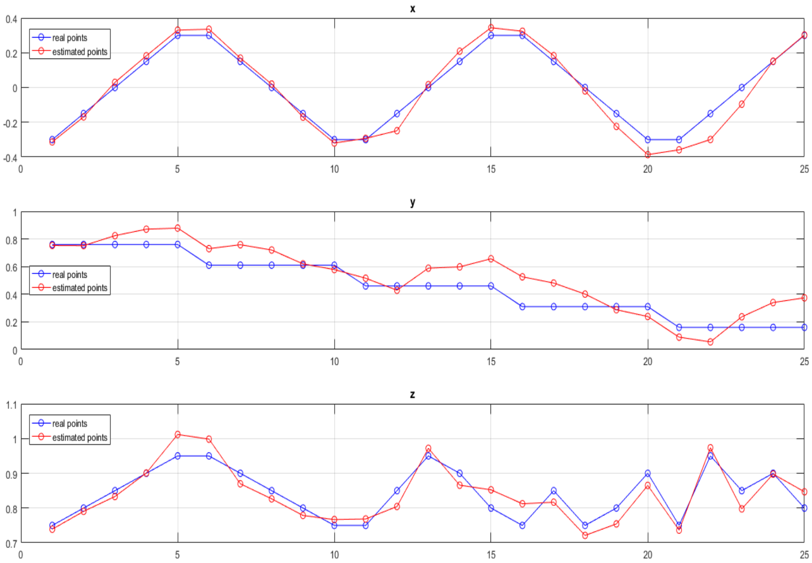


Figure 6.9: Difference between the 25 points real positions (in blue) and their estimated positions using visual data only along the x, y and z directions (in red).

6.2.3 IMUs + Cameras reconstruction

The third and last approach consists in reconstructing the human upper body fusing together the measurements provided by the inertial sensors and the data provided by the cameras. The procedure is explained in detail in chapter 4.

6.2.3.1 Short movement

Also in this case, the three-dimensional reconstruction of the human shoulders and arms at the 4 moments in which the test subject touches the grid's 4 corners is plotted in figure

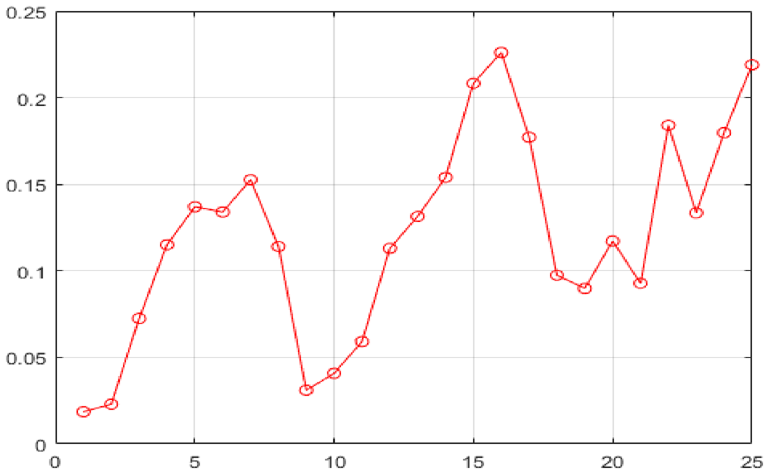


Figure 6.10: The error evolution in time using visual data only. On the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis the norm of the error vector is plotted.

6.11a. As usual, the thicker arm is the right one.
The calculated ICP error between the estimated and the real point clouds obtained after the ICP minimization procedure in this case is:

$$E_{ICP} = 4.74\ cm$$

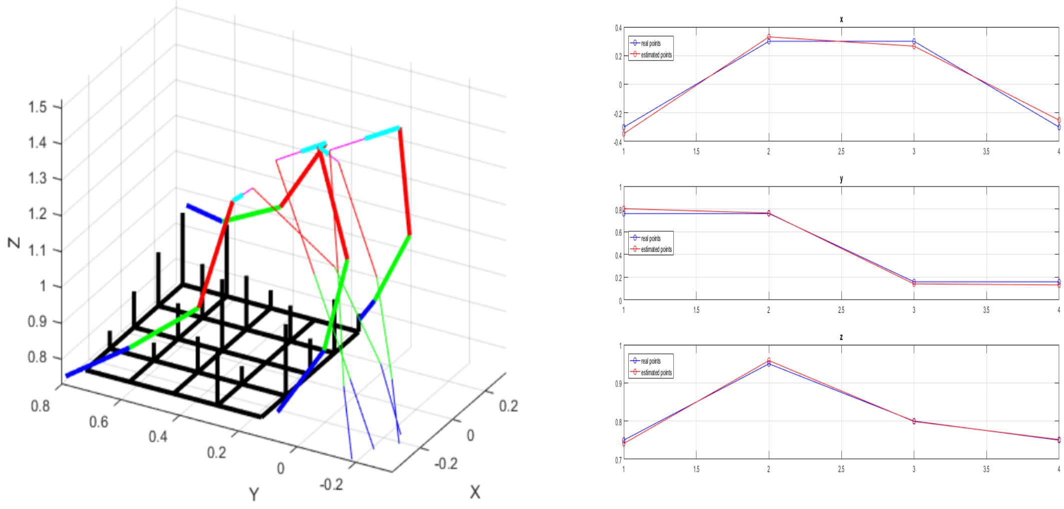
Figure 6.11b shows the difference between the real x,y,z coordinates of each corner in the global reference frame (in blue) and the estimated x,y,z coordinates of each corner obtained after the ICP minimization procedure (in red).

6.2.3.2 Long movement

For what concerns the long experiment, the calculated ICP error between the estimated and the real point clouds obtained after the ICP minimization procedure is:

$$E_{ICP} = 6.69\ cm$$

Figure 6.12 shows the difference between the real x,y,z coordinates of each stick's tip in the global reference frame (in blue) and the estimated x,y,z coordinates of each stick's tip obtained after the ICP minimization procedure (in red).
The error evolution in time obtained using a Kalman fusion algorithm of inertial and cameras measurements can be observed in figure 6.13. On the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis the norm of the error vector is plotted.



(a) 3D reconstruction of the human upper body at the 4 moments in which the test subject touches the 4 corners of the grid using Kalman fusion.

(b) Difference between the 4 corners real positions (in blue) and their estimated positions using Kalman fusion along the x, y and z directions (in red).

Figure 6.11: Three-dimensional reconstruction of the test subject touching the 4 corners of the grid and difference between the 4 corners real and estimated positions in the global reference frame, obtained using Kalman fusion of inertial and cameras measurements.

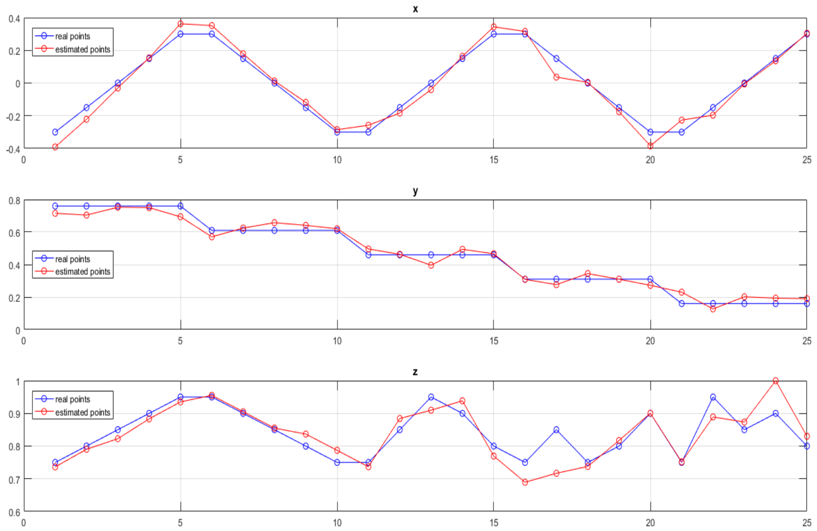


Figure 6.12: Difference between the 25 points real positions (in blue) and their estimated positions using Kalman fusion along the x, y and z directions (in red).

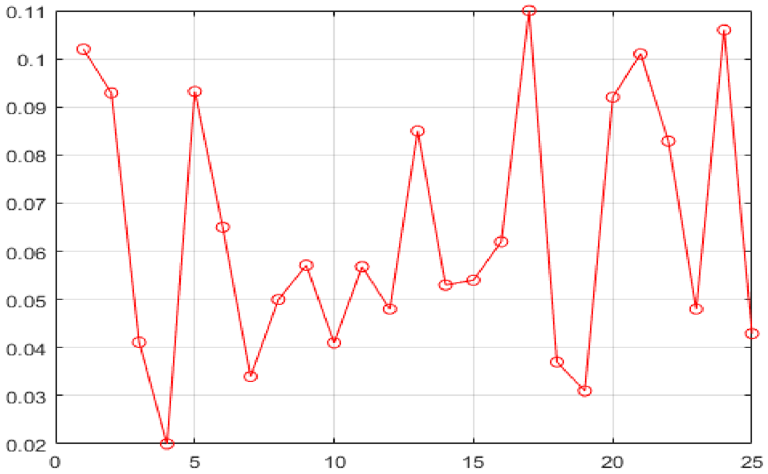


Figure 6.13: The error evolution in time using Kalman fusion. On the x-axis there are the 25 points that the test subject touches sequentially, while on the y-axis the norm of the error vector is plotted.

6.2.4 Results Comparisons

For better understanding, the error accumulations in the three long experiments resulting from reconstructing the human upper body using only IMUs, only vision and Kalman fusion of the two types of sensors are compared in figure 6.14.
For the purpose of confronting them more easily, table 6.1 summarizes the mean ICP errors of both short and long experiments obtained after the ICP minimization procedure for the three reconstruction methods explored in this thesis work.

	IMUs only	Vision only	Kalman fusion
Short experiment	6.72 cm	6.05 cm	4.74 cm
Long experiment	22.57 cm	12.08 cm	6.69 cm

Table 6.1: Mean ICP errors of short and long experiments obtained after the ICP minimization procedure for the three reconstruction methods (IMUs only, vision only, Kalman fusion of IMUs and vision measurements).

Finally, in order to give the readers a more intuitive idea of the fusion algorithm workflow, one camera frame taken at the time instant at which the test subject is touching the tip of one of the 25 sticks, the corresponding OpenPose output and the corresponding three-dimensional reconstruction obtained using the Kalman fusion algorithm of inertial and vision data are shown in figures 6.15, 6.16 and 6.17, respectively.

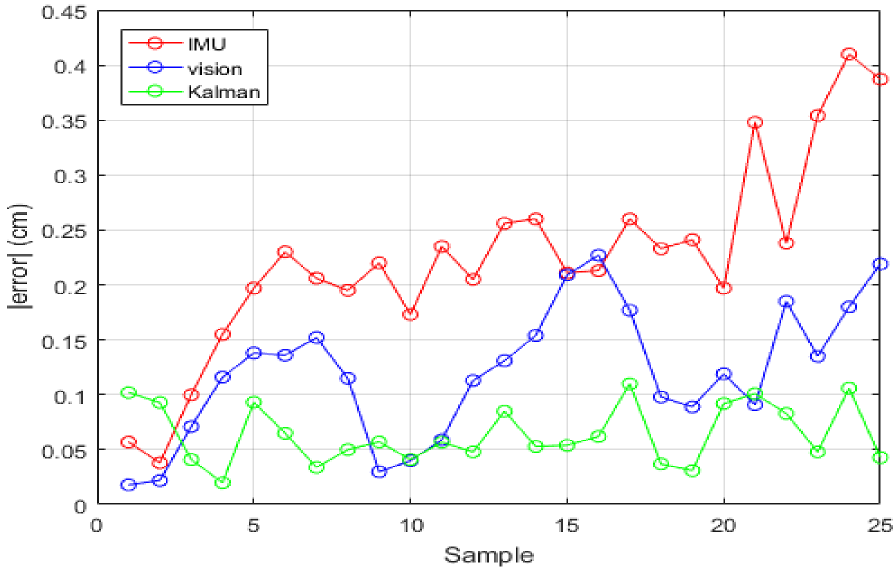


Figure 6.14: Comparison between the error accumulation in the long experiment using IMU sensors only (in red), vision only (in blue) and Kalman fusion (in green).

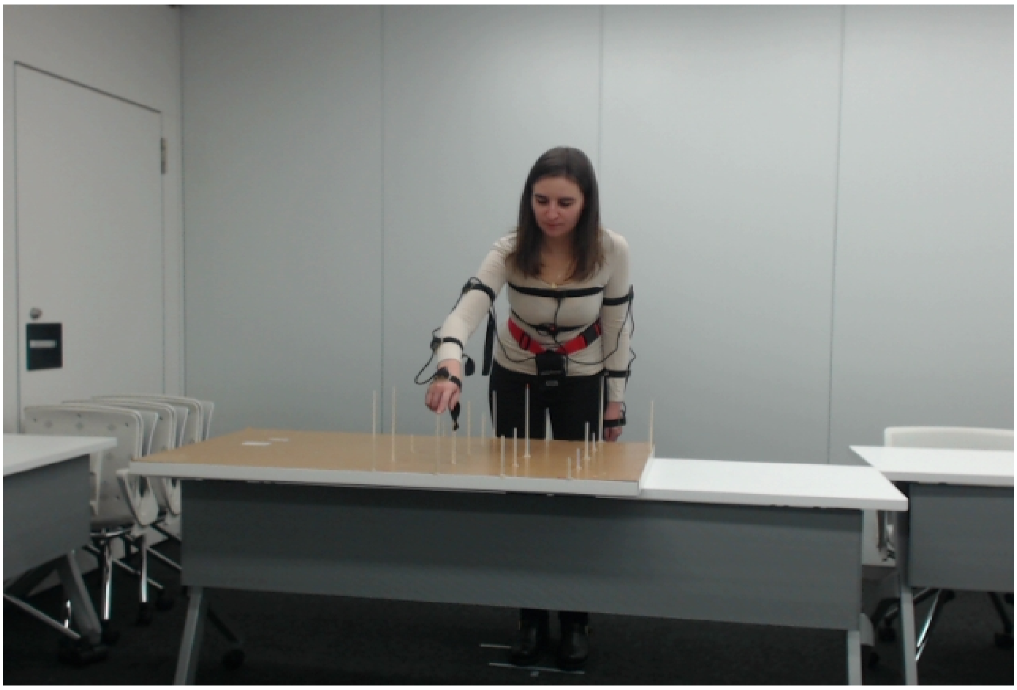


Figure 6.15: The camera frame taken at the time instant at which the test subject is touching the tip of the stick number 4 of the board.

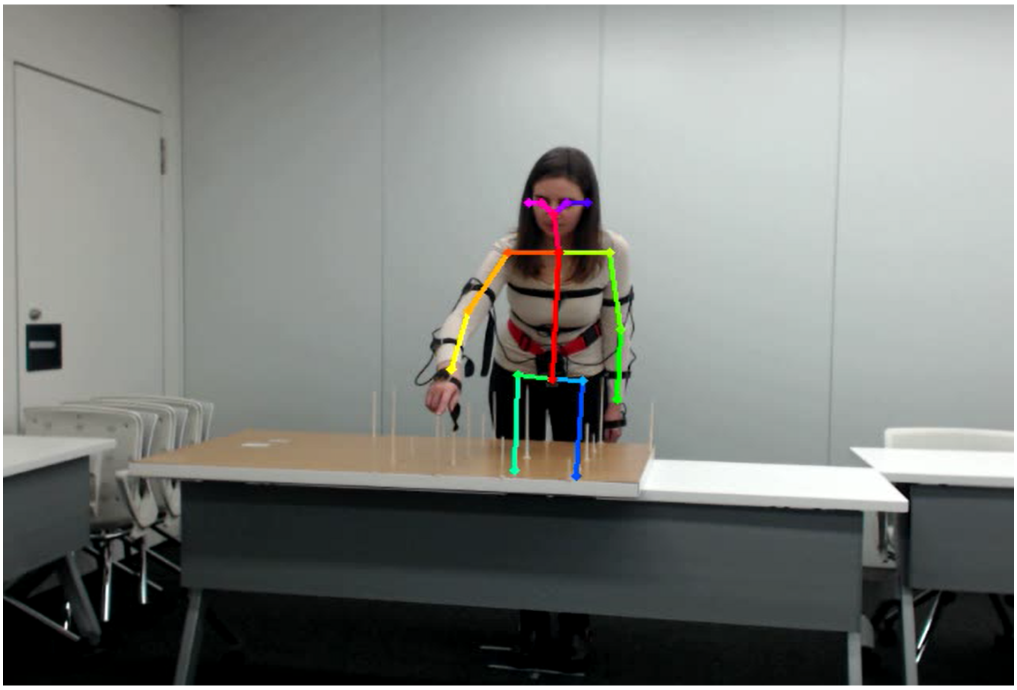


Figure 6.16: The corresponding OpenPose output, in which the joints and limbs are identified.

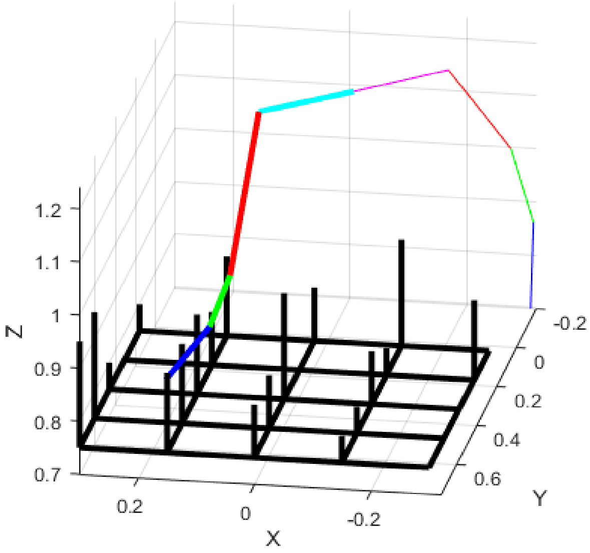


Figure 6.17: The corresponding three-dimensional reconstruction obtained using the Kalman fusion algorithm of inertial and vision data.

Chapter 7

Discussion

Looking at the experimental results, a first consideration clearly emerges: when the human upper body is reconstructed using only inertial sensors, there is an unavoidable drift error. As time goes by, the estimated positions tend to diverge a bit from the real positions of the three-dimensional grid. This trend is particularly evident in the y direction of figure 6.6 and in figure 6.7. This drift error affecting IMU sensors has been reported also by other researchers. For example, Corrales and Candelas in [22] compared the actual displacement of a person at different distances with the displacement obtained from their motion capture system and they obtained a maximum error of 66.04 cm over a distance of 200 cm, of 69.54 cm over a distance of 300 cm and of 64.23 cm over a distance of 400 cm. In some cases, the resulting error was more than 30% of the actual distance. Since this drift error was too high for industrial purposes, they had to add an additional UWB localization system to their framework. Also Zheng et al. in [14] measured the positions of hand and feet during motion along specific trajectories. They first let the tip of one hand move along the boundary of a $50cm \times 50cm$ board and obtained a maximum error of 4 cm in one direction. Then they tested walking along a straight line 4.71 m long, turning around and walking back to the starting point and found that the trajectory measured by the system was drifted of a distance around 0.4 m. The accuracy of each reconstruction method is especially evident when observing the overall error estimated with the ICP algorithm for the longer experiment, in which the test subject touched the tips of all 25 sticks of the grid. E_{ICP} is particularly high for the inertial reconstruction, reaching 22.57 cm, which is more that 30% of the actual board's dimension. This result is comparable with the ones obtained by Ramon and Candelas and by Zheng et al.

When dealing only with the vision system, the obtained errors depend mainly on the position of the subject with respect to the two cameras and on the performance of OpenPose. The error space for the stereoscopic system is notoriously non uniform and this leads to a lack of accuracy in the three-dimensional reconstruction when the subject's position is not favourable for triangulation, for instance when a limb is straight along the optical axis of the camera. Moreover, the joints positions found by OpenPose are generally accurate, but mistakes of a few pixels are absolutely normal and can result in an error of several centimeters after the triangulation is performed. As shown in figure 6.9, this leads to an error still particularly consistent in the y direction of the global reference system, which is the depth of the scene observed by the two cameras. Figure 6.10 shows instead that the error is not affected by drift; in fact, it can vary significantly over time, but this is somehow expected and clearly related to the position of the sticks rather than to the time elapsed since the start of the experiment. Moving from left to right the final joints reach indeed positions that are less favourable for triangulation, while in the left part of the experimental set such a condition is more advantageous. The ICP error in the vision-only framework decreases to 12.08 cm, showing a considerable improvement with respect to the only IMUs case.

When fusing inertial and vision data, we observe a clear limitation of the errors along all three directions (x,y,z). This leads to a good matching between the estimated points and the real points of the grid, which is especially noticeable, with respect to the previous two methods, in the y direction of figure 6.12. Moreover, figure 6.13 shows that the norm of the error vector is small and quite constant in time. When fusing inertial and vision data, the ICP error is decisively improved, reaching 6.69 cm. This result is slightly better than the one obtained by Trumble et al. in [30], whose proposed approach tested on walking, acting and freestyle performances presents a mean of the average per joint errors equal to 7 cm. However, differently from this thesis work, they used eight calibrated full HD video cameras recording at 60 Hz and they fused IMU data with a fully connected fusion layer. The result obtained in this thesis is comparable with the ones obtained by Malleson et al. in [31], and the outcome is even more remarkable when taking into account that they used both more IMU sensors and more cameras with respect to this thesis setup. They analyzed various indoor motions including walking, acting and freestyle, obtaining an average position error of 6.2 cm using thirteen IMUs and eight cameras, degrading slightly to 6.8 cm using four cameras. When using only six IMU sensors, instead, they obtained larger errors, namely 9.1 cm when using eight cameras and 14.2 cm when using four cameras. Von Marcard et al. in [32] obtained a mean 3D joint position error between 3.8 cm and 5.2 cm for their hybrid tracker when evaluating a set of walking and jogging sequences, which is a better result than the one obtained in this work. However, differently from the setup used in this thesis, they used a set of seven synchronized RGB-video cameras working at a resolution of 800 × 600 pixels, at a frame rate of 50 Hz and having orthogonal viewing directions to the scene, which provides more detailed 3D information with respect to a simple stereo setup. Moreover, they adopted a background subtraction method based on a pixel-wise Gaussian model to generate body silhouettes and ten IMU sensors to record the limbs orientation (five of them are used for tracking, the other five for validation). Obviously, when more cameras are used the ambiguities in the silhouettes decrease and the orthogonal positioning of the cameras leads to a better inference of the limb positions and orientations orthogonal to the viewing direction of the additional cameras.

A more effective comparison of the error trends of the three approaches experimentally tested is given by figure 6.14, where the error for the long experiment is plotted. Here, the accumulation of the method based on pure inertial sensors, due to the well-known drift error affecting IMU sensors, can be clearly appreciated. For the vision system only, drift hardly appears, even though error still presents sudden changes, reaching quite high values at some points. Finally, when using the Kalman fusion algorithm, no error accumulation is reported, since the norm of the error is limited and its trend is clearly bounded in time.

Chapter 8

Conclusions and Possible Applications

Collaborative robotics is one of today's major challenges in the robotics field. Cobots are intended to work alongside humans and to directly engage with them in a shared space, for social purposes or in industrial environments. Even if the performance requirements in collaborative robotics are looser than in industrial robotics, cobots are still characterized by incredibly complex specifications, which are still far from being satisfied by present-day industrial systems. In particular, the main challenge consists in respecting the strict accuracy requirements needed to ensure human safety, which leads to the need to localize the human operators in the robotic workplaces in real-time. In different terms, robots should be provided with a good spatial perception of the robotic workplace and with the ability to make reasonable predictions on the human movements, also in case of obstacles and occlusions, which is a very challenging task. Researchers have been using different methods and different kinds of sensors to deal with this problem, such as lasers, ultrasounds, vision systems, depth imaging technology or Inertial Measurement Units (IMUs). Since all these sensors presented some drawbacks which made the task of accurately reconstructing the human skeleton critical, for example being affected by occlusions, being restricted to a limited field of view or suffering from a consistent drift error, researchers started exploring other options, like the integration of the measurements coming from different types of sensors in order to exploit the advantages of each one of them and, at the same time, to compensate each sensor's drawbacks. In the literature there are many examples of fusion of Kinect and stereo vision, IMUs and Kinect, lasers and inertial sensors, IMUs and UWB technology, inertial sensors and GPS or IMUs and vision systems.

This thesis work focuses on the integration of inertial measurement units and a stereo vision system using a Kalman filter-based fusion algorithm, with the aim of localizing the human operator in the robotic workspace as accurately as possible. This problem is dealt with in a quantitative way, meaning that the performed experiments are designed with the purpose of evaluating the accuracy of the applied reconstruction methodology and to verify if such a system can be effectively employed in collaborative robotics applications.

Experimental results show that the fusion of inertial and vision data improved remarkably the accuracy of the human upper body reconstruction with respect to a vision-only or an IMUs-only approach. Using the proposed hybrid system, the presence of a human operator in a robotic workspace can be detected with an accuracy of some centimeters, which is comparable or in some cases even better than the results obtained by other researchers, who often used much more expensive equipment. The system described in this thesis work employs only two commercial cameras observing the scene and eight inertial sensors strapped to the operator's body. It is lightweight, cheap and it exploits the advantages of both types of sensors: on one hand, the presence of inertial sensors allows for the field of view not to be limited to the one of the two cameras, for the reconstruction to work also in the presence of occlusions in the workspace and for the system to work at a high data rate; on the other hand, every time a camera frame is available the greater accuracy of the vision system allows to correct the error in the joints positioning coming from the IMU measurements and to fix therefore the consistent drift error which affects inertial sensors especially during long experiments. In conclusion, such a system can be effectively employed for human localization in a robotic workspace for collaborative robotics applications.

The results achieved by this thesis could be of great interest not only for direct interaction tasks between humans and robots, but also in the characterization of advanced robotics cells. Today the characterization is performed in a kinematic way, by using different kinds of sensors to analyze the robot trajectories and determine the risk areas in which the robot and the human operators could work simultaneously. The robot is then programmed to slow down or to completely stop moving when the user enters such areas. With reference to figure 8.1, some researchers [44] proposed to divide the robotic cell into smaller zones: the red zone (detection zone), in which the robot motion is cancelled when someone steps into it and the yellow zone (warning zone), in which the robot speed is reduced to the safe value of 250 mm/s when someone steps into it, in order to prevent the complete stop of the production line. However, when the human and the robot have to work in a very small area, it is necessary to apply techniques limiting the power and the force of the robot, in parallel to its reduced speed. This safety strategy requires capacitive robotic skin that allows the robot to detect contact with the human in real-time or pressure sensitive floor mats that are capable of tracking the position of the human. Of course different safety strategies can be envisaged; for example, researchers proposed the definition of suitable comfort zones around the operator and the robot [45]. Depending on the value of the frame-by-frame distance between the two zones with respect to two thresholds, the system can be either in a safe, warning or dangerous situation, and consequently the robot will respectively continue to move at its normal speed, slow down or stop completely. A further strategy monitors instead the safety of the operator by evaluating the time instants in which he crosses some virtual barriers which delimit the three zones: the safe zone (where the human can move safely because the robot cannot reach him), the warning zone (where the contact between human and robot can happen), and the danger zone (where the robot works and can easily hit the operator).

The innovative idea brought by this thesis work is that spatial perception can be more than a simple detection of (fixed or moving) safety zones. Looking at the human body as a 3D skeleton, common trajectories in the workspaces can be gradually learnt by the robot during a cell characterization phase. Using the inertial sensors data, it is possible to train neural networks and machine learning algorithms to identify and foresee the human operator's accelerations and velocities during the execution of a specific movement. In the

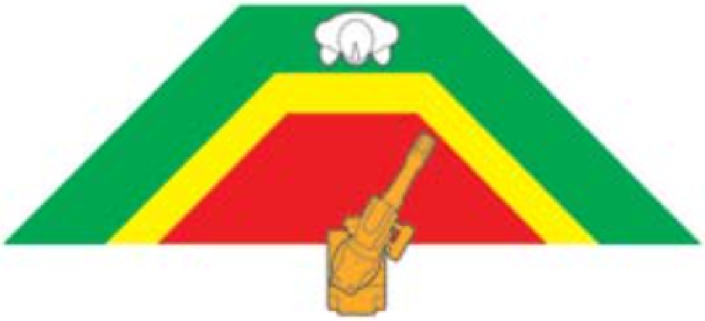


Figure 8.1: Safety zones for human-robot collaboration with speed and separation monitoring technology (Courtesy of ABB Robotics). The green one is the safe zone, the yellow one is the warning zone and the red one is the danger zone.

training phase the pure vision data is compared to the fusion data, with the final goal of learning the more accurate trajectories, i.e. those reconstructed with the fusion algorithm, starting from vision data only. At this point, it will be possible to eventually disregard the IMU sensors, which are often bulky and can sometimes limit the operator's freedom of motion, in favour of a pure three-dimensional vision reconstruction, where only cameras, which are cheap and very common, are used.

In summary, this work will hopefully be a starting point to better investigate the concept of "spatial intelligence" for a robotic system, taking into consideration real environments characterized by people in motion and rigorous safety constraints. Interestingly, this concept of spatial intelligence can hardly be separated from the great need for "adaptive behaviors" that we expect from advanced robots; moreover, it cannot be separated from a basic "learning ability" which should take into account the reliability of various sensory systems and the correct evaluation of repeated trial and error phases. In this perspective, the fusion strategy adopted by this thesis could be easily extended to additional sensory systems and easily integrated with state-of-the-art learning strategies.

Bibliography

[1] James E Colgate and Michael A Peshkin. *Cobots*. US Patent 5,952,796. Sept. 1999.

[2] Keisuke Okada, Jae Hoon Lee, and Shingo Okamoto. “People tracking by collaboration between a mobile robot and static laser scanner”. In: *Proceedings of the International MultiConference of Engineers and Computer Scientists*. Vol. 2. 2014.

[3] Stefan Holban et al. “3D RECONSTRUCTION OF OBJECTS FROM ULTRASOUND IMAGES”. In: ().

[4] Fabio Remondino and Andreas Roditakis. “3D reconstruction of human skeleton from single images or monocular video sequences”. In: *Joint Pattern Recognition Symposium*. Springer. 2003, pp. 100–107.

[5] Gareth Loy et al. “Monocular 3d reconstruction of human motion in long action sequences”. In: *European Conference on Computer Vision*. Springer. 2004, pp. 442–455.

[6] Yen-Lin Chen and Jinxiang Chai. “3D Reconstruction of Human Motion and Skeleton from Uncalibrated Monocular Video”. In: *ACCV*. 2009.

[7] Riza Alp Guler and Iasonas Kokkinos. “Holopose: Holistic 3d human reconstruction in-the-wild”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 10884–10894.

[8] Meiyin Liu, SangUk Han, and SangHyun Lee. “Tracking-based 3D human skeleton extraction from stereo video camera toward an on-site safety and ergonomic analysis”. In: *Construction Innovation* 16.3 (2016), pp. 348–367.

[9] Jungong Han et al. “Enhanced computer vision with microsoft kinect sensor: A review”. In: *IEEE transactions on cybernetics* 43.5 (2013), pp. 1318–1334.

[10] Jamie Shotton et al. “Real-time human pose recognition in parts from single depth images”. In: *CVPR 2011*. Ieee. 2011, pp. 1297–1304.

[11] Dimitrios S Alexiadis et al. “Evaluating a dancer’s performance using kinect-based skeleton tracking”. In: *Proceedings of the 19th ACM international conference on Multimedia*. 2011, pp. 659–662.

[12] Alessandro Filippeschi et al. “Survey of motion tracking methods based on inertial sensors: A focus on upper limb human motion”. In: *Sensors* 17.6 (2017), p. 1257.

[13] Daniel Roetenberg, Henk Luinge, and Per Slycke. “Xsens MVN: Full 6DOF human motion tracking using miniature inertial sensors”. In: *Xsens Motion Technologies BV, Tech. Rep* 1 (2009).

[14] Y. Zheng, K. Chan, and C. C. L. Wang. “Pedalvatar: An IMU-based real-time body motion capture system using foot rooted kinematic model”. In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Sept. 2014, pp. 4130–4135. DOI: 10.1109/IROS.2014.6943144.

[15] W. Kong et al. “Development of a real-time IMU-based motion capture system for gait rehabilitation”. In: *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. Dec. 2013, pp. 2100–2105. DOI: 10.1109/ROBIO.2013.6739779.

[16] Minmin Zhang. “Multi-sensor inertial measurement system for analysis of sports motion”. PhD thesis. University of Pittsburgh, 2015.

[17] Z. Lin et al. “Objective Skill Evaluation for Laparoscopic Training Based on Motion Analysis”. In: *IEEE Transactions on Biomedical Engineering* 60.4 (Apr. 2013), pp. 977–985. DOI: 10.1109/TBME.2012.2230260.

[18] W. Jia et al. “3D image reconstruction and human body tracking using stereo vision and Kinect technology”. In: *2012 IEEE International Conference on Electro/Information Technology*. May 2012, pp. 1–4. DOI: 10.1109/EIT.2012.6220732.

[19] François Destelle et al. “Low-cost accurate skeleton tracking based on fusion of kinect and wearable inertial sensors”. In: *2014 22nd European Signal Processing Conference (EUSIPCO)*. IEEE. 2014, pp. 371–375.

[20] Yushuang Tian et al. “Upper limb motion tracking with the integration of IMU and Kinect”. In: *Neurocomputing* 159 (2015), pp. 207–218.

[21] Mohammad Safeea and Pedro Neto. “Minimum distance calculation using laser scanner and IMUs for safe human-robot interaction”. In: *Robotics and Computer-Integrated Manufacturing* 58 (2019), pp. 33–42.

[22] Juan Antonio Corrales, FA Candelas, and Fernando Torres. “Hybrid tracking of human operators using IMU/UWB data fusion by a Kalman filter”. In: *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2008, pp. 193–200.

[23] Yuan Xu et al. “Adaptive robust INS/UWB-integrated human tracking using UFIR filter bank”. In: *Measurement* 123 (2018), pp. 1–7. ISSN: 0263-2241. DOI: https://doi.org/10.1016/j.measurement.2018.03.043. URL: http://www.sciencedirect.com/science/article/pii/S0263224118302203.

[24] Matthew Brodie, Alan Walmsley, and Wyatt Page. “Fusion motion capture: a prototype system using inertial measurement units and GPS for the biomechanical analysis of ski racing”. In: *Sports Technology* 1.1 (2008), pp. 17–28.

[25] Yahui Liu et al. “An innovative information fusion method with adaptive Kalman filter for integrated INS/GPS navigation of autonomous vehicles”. In: *Mechanical Systems and Signal Processing* 100 (2018), pp. 605–616.

[26] Chen Chen, Roozbeh Jafari, and Nasser Kehtarnavaz. “A survey of depth and inertial sensor fusion for human action recognition”. In: *Multimedia Tools and Applications* 76.3 (2017), pp. 4405–4425.

[27] Gabriel Nützi et al. “Fusion of IMU and vision for absolute scale estimation in monocular SLAM”. In: *Journal of intelligent & robotic systems* 61.1-4 (2011), pp. 287–299.

[28] Korbinian Schmid and Heiko Hirschmüller. "Stereo vision and IMU based real-time ego-motion and depth image computation on a handheld device". In: *2013 IEEE International Conference on Robotics and Automation*. IEEE. 2013, pp. 4671–4678.

[29] Timo von Marcard et al. "Recovering accurate 3d human pose in the wild using imus and a moving camera". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 601–617.

[30] Matthew Trumble et al. "Total Capture: 3D Human Pose Estimation Fusing Video and Inertial Sensors." In: *BMVC*. Vol. 2. 2017, p. 3.

[31] Charles Malleson et al. "Real-time full-body motion capture from video and imus". In: *2017 International Conference on 3D Vision (3DV)*. IEEE. 2017, pp. 449–457.

[32] Timo Von Marcard, Gerard Pons-Moll, and Bodo Rosenhahn. "Human pose estimation from video and imus". In: *IEEE transactions on pattern analysis and machine intelligence* 38.8 (2016), pp. 1533–1547.

[33] James Diebel. "Representing attitude: Euler angles, unit quaternions, and rotation vectors". In: *Matrix* 58.15-16 (2006), pp. 1–35.

[34] Alexander Rampp et al. "Inertial sensor-based stride parameter calculation from gait sequences in geriatric patients". In: *IEEE transactions on biomedical engineering* 62.4 (2014), pp. 1089–1097.

[35] S Sessa et al. "Walking assessment in the phase space by using ultra-miniaturized Inertial Measurement Units". In: *2013 IEEE International Conference on Mechatronics and Automation*. IEEE. 2013, pp. 902–907.

[36] Weisheng Kong et al. "Anatomical calibration through post-processing of standard motion tests data". In: *Sensors* 16.12 (2016), p. 2011.

[37] Eduardo Palermo et al. "Experimental evaluation of accuracy and repeatability of a novel body-to-sensor calibration procedure for inertial sensor-based gait analysis". In: *Measurement* 52 (2014), pp. 145–155.

[38] Tobias Zimmermann, Bertram Taetz, and Gabriele Bleser. "IMU-to-segment assignment and orientation alignment for the lower body using deep learning". In: *Sensors* 18.1 (2018), p. 302.

[39] Olivier Faugeras and OLIVIER AUTOR FAUGERAS. *Three-dimensional computer vision: a geometric viewpoint*. MIT press, 1993.

[40] Zhe Cao et al. "OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields". In: *arXiv preprint arXiv:1812.08008* (2018).

[41] Rudolph Emil Kalman. "A new approach to linear filtering and prediction problems". In: (1960).

[42] Fabio Scibona. "Sistema di navigazione basato sul filtro di Kalman per una piattaforma d'assetto a tre assi: sviluppo software, hardware e test". In: (2015).

[43] P. J. Besl and N. D. McKay. "A method for registration of 3-D shapes". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14.2 (Feb. 1992), pp. 239–256. ISSN: 1939-3539. DOI: 10.1109/34.121791.

[44] George Michalos et al. "Design considerations for safe human-robot collaborative workplaces". In: *Procedia CIrP* 37 (2015), pp. 248–253.

[45] Simone Pasinetti et al. "Development and characterization of a safety system for robotic cells based on multiple Time of Flight (TOF) cameras and point cloud analysis". In: *2018 Workshop on Metrology for Industry 4.0 and IoT*. IEEE. 2018, pp. 1–6.