

Visual-inertial prediction of terrain-specific human locomotion modes

Ali Hussein Abdulzahra AL-DABBAGH

Thesis submitted in partial fulfillment of the requirements for the degree of
Docteur en sciences de l'ingénieur et technologie (Ph.D. thesis)

October 2022

Examining committee

Pr.	H. Jeanmart	UCLouvain	<i>Chairperson</i>
Pr.	R. Ronsse	UCLouvain	<i>Supervisor</i>
Pr.	B. Dehez	UCLouvain	
Pr.	C. De Vleeschouwer	UCLouvain	
Pr.	R. Detry	KU Leuven	
Pr.	E. Aksoy	Halmstad University	

Acknowledgments

This thesis marks the achievement of almost four years of a Ph.D. that were spent at the Mechatronics, Electrical Energy, and Dynamic Systems (MEED) department at the Université Catholique de Louvain (UCLouvain) in Belgium.

First, I would like to express my sincere gratitude to my supervisor, Renaud Ronse, for guidance, encouragement, scientific expertise, and the highest level of standard regarding publication writing. I consider myself extremely lucky to have a supervisor who cared a lot about my work, and who responded to my questions and queries so promptly through individual and weekly lab meetings.

Also, special thanks to Thierry Daras and Benoît Herman for assisting me in building the hardware parts of my project. On the other hand, I would like to thank Catherine Bauwens and Nathalie Sergoigne for their administrative support before and during my Ph.D. journey.

I also wish to express my gratitude to all colleagues and friends in MEED for accepting to participate in many experiments that I have conducted during my Ph.D. to evaluate the system's performance. Indeed, I feel so proud for being part of such nice and friendly teammates who are always ready to help in many ways.

Further, I want to express my gratitude to the Ph.D committee for their time and effort in providing me with insightful feedback to improve this thesis and take it to a higher level.

Last but not least, I would like to thank my wife for her constant and unconditional support during my whole academic studies in Australia and Belgium.

Funding

This work was supported by the Conseil de l'action internationale of UCLouvain and the European Community's H2020 Research and Innovation Programme under grant number 731931 (the CYBERLEGsPlusPlus collaborative project)

Abstract

The exponential growth of technology opened the door for a new generation of locomotion assistive devices (e.g. prostheses and exoskeletons) to emerge. The latest generation is lighter weight due to the emergence of smart and composite materials, and advanced actuators and sensors. Controlling these devices to compensate for limb loss or walking disorder requires the generation of various gait patterns for different locomotion tasks like stairs ascending, descending, and level ground walking. However, this requires that these devices be able to detect the actions and intentions of their human user when traversing from one type of terrain to another. Such transition requires switching to the new task in a timely manner; otherwise, the user is more vulnerable to falls and injuries. Typical solutions to detect a new locomotion task are based either on sensors that measure the user's kinematics or on a voluntary action of the user, such as tilting their chest or pushing a button. While the former suffers from a delay, the latter provides an unnatural way of behaving during locomotion. Recently, alternative methods have been proposed that use wearable ranging sensors or camera. However, the implemented approaches for terrain identification with such sensors have been only validated in single indoor and/or outdoor path(s). Also, the system detection ability in a path being partially occluded by other walkers has not been addressed yet. Last but not least, the detection results are obtained based on a pre-specified region of interest in front of the user, without applying extra processing to predict the terrain information for several upcoming steps.

In this thesis, we introduce a computer vision-based terrain detection algorithm that relies on a depth camera attached to the user's chest. The camera, therefore, could capture the scene in front of the user in a point cloud data format, while the algorithm applies several steps on the captured data in order to achieve environmental feature extraction. The features are passed to a trained classifier that predicts so-called locomotion affordances. The algorithm further processes the detected affordances to obtain the terrain type and its geometrical features for three steps in advance with the adaptation to the instantaneous user's step size. Additionally, the system can provide an online estimate of the walking stride length, speed, and turning angle based on a single inertial sensor that is built within the depth camera. The estimated speed and turning angle pave the way to an activity tracking method that could be beneficial to evaluate the assistive device behavior across various environmental terrains without the need for dedicated equipment.

The algorithm was validated with 8 participants in several indoor and outdoor

paths with a variety of terrain types. The validation process consisted, also, in conducting the experiments with clear and partially occluded path conditions. The system accomplishes remarkable results with a grand average detection accuracy above 90% over all locomotion modes with clear path conditions. Moreover, it demonstrates the ability to recognize various terrain types even with partially occluded path conditions at least for a single step in front of the user. Finally, the estimated geometrical features of most terrain types were closed to real observed values, even with partially covered paths. Similar low error rates were observed for the walking speed and step size estimation approaches when evaluated on several participants walking across paths designed for that purpose.

The full package which consists of terrain detection for multiple steps, geometrical features, speed, and turning angle estimations can provide a valuable tool for a wide variety of disabled people including amputees with an active prosthesis, visually impaired, or blind people who face special locomotion difficulties in daily life.

Keywords: Terrain detection, Locomotion Assistance, Computer Vision, Wearable Sensors, Machine Learning, Prosthesis, Exoskeleton, Wearable Vision, Assistive Technology, Depth Vision, Stairs Detection, Ramps Detection, Robotics Vision.

Contents

1	Introduction	1
1.1	Context	1
1.2	Scientific questions	6
1.3	Structure of the thesis	7
1.4	Publications	8
2	Terrain detection systems for locomotion assistance: a review	9
2.1	Introduction	9
2.2	Systems with proprioceptive sensors only	13
2.2.1	EMG sensors	13
2.2.2	Capacitive sensors	14
2.2.3	Mechanical sensors	15
2.2.4	Fusion of Neuromuscular and Mechanical sensors	16
2.2.5	Summary	17
2.3	Systems with at least one exteroceptive sensor	20
2.3.1	Ranging Sensors	21
2.3.2	Depth Sensors	22
2.3.3	Fusion of RGB and Depth Sensors	29
2.3.4	Summary	29
2.4	Discussion	30
2.5	Conclusion	34
3	Early terrain detection system implementation	35
3.1	Introduction	35
3.2	Terrain detection algorithm	36
3.2.1	Downsampling and Cropping	36
3.2.2	Features extraction	37
3.2.3	Features transformation	38
3.2.4	Classification tree	39
3.3	Experimental protocol	39
3.3.1	Software	39
3.3.2	Hardware	39

3.3.3	Experiments	40
3.3.4	Accuracy	41
3.4	Results and discussion	42
3.5	Conclusion	45
4	Terrain detection system for multiple steps with a fixed step size	47
4.1	Introduction	47
4.2	Terrain detection algorithm	48
4.2.1	Affordances detection	49
4.2.2	Terrains estimation for three steps	52
4.3	Experimental validation	54
4.3.1	Methods	54
4.3.2	Protocol	54
4.3.3	System Training and evaluation	56
4.3.4	System performance estimation for the three steps	57
4.4	Results	58
4.4.1	Classifiers performance	58
4.4.2	Locomotion-mode classification	60
4.4.3	Terrains geometrical features	60
4.4.4	Frame processing rate	63
4.5	Discussion	63
4.6	Conclusion	68
5	Walking speed, cadence, step size and turning estimation	69
5.1	Introduction	69
5.2	Walking speed estimation	69
5.3	Cadence and Step size estimation	71
5.4	Turning angle estimation	72
5.5	Experimental methods	73
5.5.1	Walking Speed and turning angle evaluation	73
5.5.2	Step size evaluation	75
5.6	Results and discussion	76
5.6.1	Walking speed estimation	76
5.6.2	Cadence and step size estimation	77
5.6.3	Turning angle estimation	82
5.7	Conclusion	82
6	Terrain Detection System for multiple steps with adaptive step size	85
6.1	Introduction	85
6.2	System overview	85
6.3	Preliminary results	87
6.4	Prediction time	88
6.5	Conclusion	90
7	General discussion and conclusion	93
7.1	Contributions	93
7.2	Discussion	94
7.3	Research questions and (partial) answers	97

7.4	Conclusion and perspectives	98
A	Experimental data of terrain detection system for several steps with classification tree	103
A.1	Experimental methods	104
A.2	Accuracy estimation	104
A.3	Results and discussion	105
A.3.1	Uncluttered environment	105
A.3.2	Cluttered environment	106
A.3.3	Discussion	109
B	Detection accuracy results	111
C	3D-Point Cloud dataset of various geometrical terrains in urban environments recorded during human locomotion	113
C.1	Dataset description	113
D	The Algorithm implementation in ROS	117
	Bibliography	121

Glossary

Terms	Definition
Accuracy	The ratio of the true positive predicted samples to the total number of samples collected within a period of time.
Affordance	According to Don Norman (2013) <i>"an affordance is a relationship between the properties of an object and the capabilities of the agent that determine just how the object could possibly be used"</i> [1]. In this thesis, we thus use the word "affordance" to capture the locomotion modes could be "afforded" by a human in a given locomotion environment.
Affordances detection	Process of determining the upcoming locomotion affordances that the agent could use in order to traverse the detected plane(s) in front of him/her.
Confusion matrix	It is used to describe the performance of a classification system [2]. In this thesis, the rows of the matrix represent the instances in the actual locomotion mode, and the columns represent the instances in the predicted locomotion mode.
Features	Pieces of information or unique properties that are contained in a waveform, 2D or 3D images. For a waveform, this information can be the minimum, maximum, mean, or number of zero crossing for a specific interval. The horizontal and vertical edges can be considered as features in the case of 2D images, while for 3D-image (point clouds) as in this thesis, the features are represented by the properties of the detected plane, such as their centroid and slope.
Features extraction	The process of extracting the properties or information from a waveform, 2D or 3D image in order to estimate the speed, or identify the upcoming terrain type.
Features transformation	Transforming the extracted features (centroid and slope) from the sensor frame to the world frame in order to identify the terrain type.
Geometrical or terrain features	This captures the plane height in case of stairs, and orientation in case of ramps or leveled ground. These are computed from the transformed-extracted features of each plane (centroid and slope).

Introduction

1

1.1 Context

Humans perform locomotion tasks with their lower limbs to commute from one place to another over different kinds of terrains [3] (see Fig. 1.1). However, this capacity can be seriously compromised with locomotion disorders such as amputation. Indeed, in Europe and US, there are around 300.000 new major lower limb amputations (above the ankle) each year [4]. This number gets much bigger if we included the rest of the world. Passive prostheses are typically used to compensate for limb loss in amputees, although inverse dynamic analysis reveals that walking humans generate active (i.e. with positive power) kinetic patterns for various types of terrains during locomotion [5]. As consequence, the development of leg prostheses switched from passive to active solutions.

The state of the art of active prostheses – or bionic limbs – is exploding nowadays, both for the upper [6] and lower extremities [7, 8]. This is mainly driven by the



Figure 1.1: stairs-ramps terrains combination that could be faced in urban environment "Photo by Leon, "Robson square vancouver stairs drawings," 2022".

maturation of miniaturized technologies for energy storage, (biocompatible) sensors, and compliant actuators [9]. Active ankle prostheses (also called “Bionic Feet”) are devices equipped with a power actuator and are thus capable of providing net positive energy over the walking cycle [10].

The development of these advanced prostheses was mostly associated with closed-loop controllers that focused on steady-state behavior (overground walking in most cases). However, many locomotion environments encountered in everyday life have different terrain features, and moving requires to transition between these environments. Indeed, adapting to different types of terrain such as uneven ground, stairs, or inclines requires that walkers continuously adapt their gait pattern. Walking on uneven surfaces, in particular, is difficult for individuals with lower limb amputation since the occurrence of falls, injuries, and fear of falling are ubiquitous [12]. On top of that, amputees tend to walk slower and with higher energy expenditure and rating of perceived exertion when exposed to a particularly challenging environment [13]. For example, an obstacle could elicit a trip if the step does not have sufficient toe clearance [14].

As consequence, active prostheses are necessary to provide disabled users with the capacity to display healthy gait patterns, and the control of such devices should be adaptive to the task being executed (walking, taking stairs, slopes, etc.). Therefore, several research labs and companies develop active robotic limbs having the capacity to reproduce such healthy gait patterns. The latest generation of lower-limb prostheses are self-powered (active), lightweight, and controlled by embedded processors to generate different locomotion patterns for different tasks [15–17] as illustrated in Fig. 1.2 (a).

Other types of active locomotion assistive devices are robotic exoskeletons which are introduced as augmentative devices for enhancing the wearer’s physical perfor-

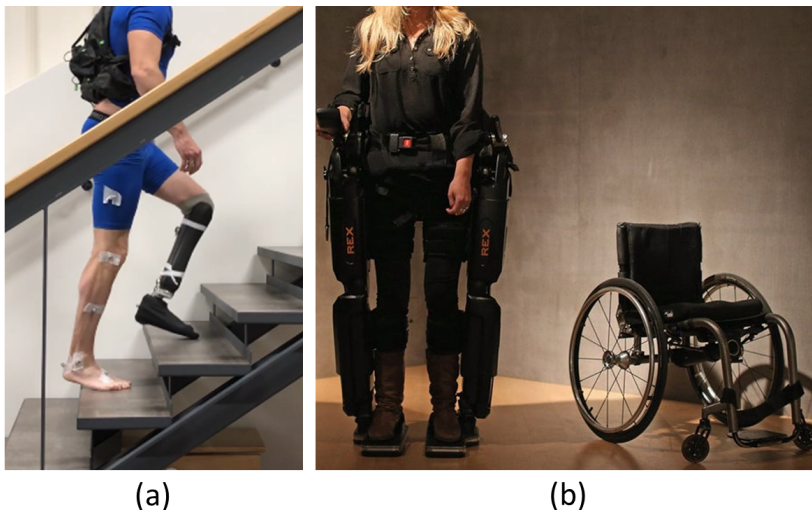


Figure 1.2: Typical active assistive devices. (a) shows an amputee going up stair with the help of an active lower limb prosthesis. (b) displays an exoskeleton worn by a disabled user to provide an active support [11].



Figure 1.3: Typical exoskeletons designed for supporting healthy people to perform heavy duty tasks such as lifting or traversing over various terrains and carrying heavy equipments [19, 20].

mance. Some of these devices are developed to assist disabled people with appropriate physical abilities for performing several locomotion tasks such as standing, walking, and stair climbing (see Fig. 1.2 (b)) [18]. Moreover, a newer version of such exoskeletons is introduced to provide support for healthy people in industry or military sectors who perform heavy-duty tasks such as lifting or carrying heavy equipment for long distances as illustrated in Fig. 1.3.

Prosthesis and exoskeletons have demonstrated their ability to provide adapted assistance for disabled and healthy people in various locomotion modes or tasks [21]. However, such systems still face several challenges regarding their interactions with human users. For instance, they usually lack the capability to timely detect the actions and intentions of the human wearer [21]. Therefore, transitions between two locomotion modes are always handled in a reactive way. Indeed, the upcoming locomotion task is either detected by sensors measuring the user's kinematics [22] and this detection is thus systematically affected by delays – or is triggered by specific and unnatural action of the user itself [23]. In the present thesis, we would like to make a breakthrough in this challenge of managing locomotion transitions, by equipping such devices with predictive sensory capacities (see Fig. 1.4). These predictions will emerge by extracting so-called locomotion affordances from the environment. Historically, the concept of sensorimotor affordances emerged in the context of ecological psychology [24], postulating that perception is economical, i.e., driven only by the perception of relevant environmental information. It then percolated to several engineering fields, including robotic grasping [25, 26], and, more recently, humanoid loco-manipulation [27, 28]. Extracting the upcoming locomotion affordances can be achieved through terrain detection systems. Such systems have been widely developed for mobile robotic applications in various environments like urban [29], off-road [30], and planetary [31] explorations. Typically, these systems also combine several sensory modalities such as cameras, IMUs, or 2D and 3D lasers [32]. Owing to the miniaturization of these components, voices raise claiming that similar

wearable terrain detection systems will be integrated into the control loop of human locomotion assistance devices [9]. Moreover, lower-limb disabled people are not the only ones who could benefit from terrain detection technologies for walking assistance. For instance, there are around 285 million visually impaired people worldwide relying on a guide dog or cane for walking [33]. Using a guide cane provides only a short-range detection capacity to the user, such that safety might be compromised in a challenging environment. Consequently, there have been recent research efforts to develop augmented feedback technologies for visually impaired people, detecting the surrounding terrains in the local environment with a sufficiently short latency to let them adjust their gait accordingly [34, 35].

Various algorithms are applied to detect different terrain types in front of the user, and these typically rely on computer-vision tools [36, 37]. Some of these systems are implemented to detect the type of terrain (non-geometrical terrains) such as sand, grass, asphalt, etc [38, 39]. Other implementations focus on geometrical terrains like stairs, ground, and/or slopes [36, 37]. It is reported that foot kinematics is influenced by ground surface type (non-geometrical terrains) for running or walking movement [8]. However, with geometrical terrains, larger gait adaption is needed. For instance, moving upstairs needs kinetic and kinematic adaptation very different from that needed for level ground walking [40]. Therefore, systems providing information about the upcoming geometrical terrains are potential of high interest to guide the device controller as a function of the intended task [41, 42].

Many recent implementations of such systems suffer from several limitations. Firstly, they provide terrain information for only one step in advance [43, 44], while detecting the upcoming terrains for several steps can lead to achieving better predictability and more natural transitions between gait patterns for disabled people. Secondly, detecting the terrain type is not always sufficient to provide an adaptive control scheme. Indeed, the geometrical features of these terrains such as step height, or ground inclination could vary from one step to another. Such feature estimation is a potential of interest for the controller of active locomotion devices like bionic prostheses or exoskeletons since their commanded kinematic or torque patterns could be finely modulated as a function of the estimated environmental features. However, such important information about the terrains is not addressed by several papers recently published [45, 46]. Moreover, the algorithm adaptability to different environments must be guaranteed to ensure the system usability in various urban contexts.

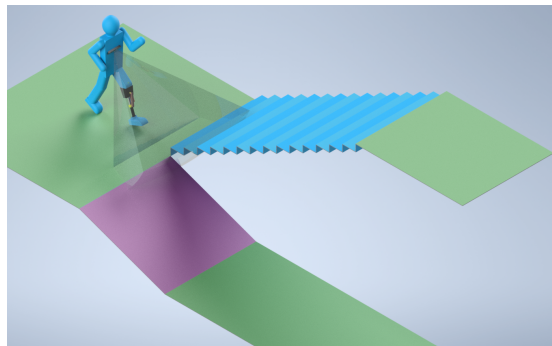


Figure 1.4: The use of predictive sensory capacities can provide an accurate estimation of upcoming affordances in complex environments.

Finally, since such systems are intended to assist disabled people in urban environments, they should, also, have the capabilities to detect the terrain even if it was partially occluded by people walking in front of the user.

Consequently, the goal of this thesis is to address these limitations regarding terrain identification for several steps, geometrical features estimation, and the capability of terrain recognition even with partial occlusion. This would deliver better assistance for active prosthesis or exoskeleton users through early switching to the newly detected locomotion mode. Furthermore, it is aimed also at equipping these devices with walking speed, cadence, step size, and turning angle estimation capabilities, again with the objective to equip assistive devices with fine adaptation capabilities. Incidentally, such pieces of information have also a potential interest for the design engineers to monitor the functioning of the assistive device on various terrains. Visually impaired people can also benefit from such a system for restoring their navigation skills. Therefore, the long-term vision behind this thesis should lead to a better quality of life for several categories of disabled people towards reaching more natural gait patterns and safer transitions over different urban terrains.

1.2 Scientific questions

There are three main scientific questions that we want to address in this thesis:

A *How can we achieve terrain detection for several upcoming steps?*

Obtaining the locomotion affordances for more than one upcoming step would give sufficient time for the prosthesis or exoskeleton to react. This mainly depends on the sensory capabilities and the applied algorithm in order to recognize the upcoming terrains. Therefore, this thesis provides an intensive review of terrain detection systems that can be used for locomotion assistance. Also, we classify these systems based on their prediction capacities. Based on our observations of this review study, we selected the right sensor for wearable applications besides implementing the algorithm to accomplish terrain identifications for more than one step. Indeed, we relied on segregating the sensor information into several regions which correspond to the upcoming steps. The amount of segregation is driven as a function of the user step size. Consequently, this thesis presents as well a step size estimation approach.

B *How can we achieve geometrical features estimation for the detected terrain in the upcoming steps?*

Such features can play an important role in tuning the torque parameters of the assistive devices (prosthesis and exoskeletons) to achieve adaptive assistance for the same terrain type with different geometrical features. Therefore, an algorithm is developed in this thesis that extracts such features from the detected affordances.

C *How and to what extent can we obtain the terrain type identification and geometrical features estimation even with a partially occluded environment?*

In the urban context, it is not guaranteed that locomotion terrains are free of other walkers. Accordingly, the implemented algorithm must show an ability to capture terrain information and features with partial occlusion. We present an algorithm that relies on 3D construction of the affordances which allows terrain identification even with partial covering of the scene in front of the user. Also, we provide an assessment method for such a scenario to evaluate the algorithm's robustness and limitation.

1.3 Structure of the thesis

The structure of the thesis follows a logical progression of our implementation of the terrain detection algorithm. Starting with an intensive review of many articles to several implementation stages within simulated and real environments. Also, the thesis covers different versions of the implemented algorithm with the conducted experiments towards reaching the final outcomes. The following paragraphs summarize the content of each chapter.

Chapter 2 explores more than one hundred papers that present various implementations of terrain detection systems that can be used for locomotion assistance. Also, it classifies these approaches based on their used sensory modalities and provides an estimation method for their prediction capacity. Interestingly, most of the collected papers share many similar procedures. Therefore, it is necessary to gather the commonly applied processes in several comprehensive figures to give ourselves and readers the opportunity of better understand the underlying purpose of each process towards reaching various locomotion mode estimations. This chapter, also, critically discusses the review methods from several aspects.

Chapter 3 introduces the early version of the terrain detection algorithm which applies several processing steps on the acquired point cloud data from the depth camera to achieve locomotion affordances recognition based on a manually tuned classification tree. The initial tests were conducted in simulated environments. After that, we designed a wearable tool to hold the depth camera to collect the data and assess the algorithm's performance in the real environment.

Chapter 4 presents the terrain detection approach with the ability of performing the terrain identification for several steps ahead of the user. The affordances firstly are recognized using a more robust classifier than the one used in the previous chapter. Then, terrain estimation for several steps approach is applied. The geometrical features of the detected terrains are also linked to each predicted step. This Chapter also illustrates algorithm evaluation methods with multiple paths and participants with clear and occluded path conditions, while the user's step sizes were considered to be fixed during each trial. Finally, the results are introduced and discussed corresponding to the state of the art.

Walking speed, cadence, step size, and turning angles estimation methodologies are introduced in Chapter 5 with their assessment methods. The purpose of the step size estimation is to predict the upcoming step in proportion to the user's stride length. The speed and turning angle could provide extra information about the assistive device behavior over multiple terrains, while the turning angle information could be used to adjust the device control parameters during the turning phase.

Therefore, Chapter 6 combines the algorithms presented in Chapter 4 and Chapter 5 to obtain the final system version which provides terrain estimation for several steps adaptive to the instantaneous user's stride length. Also, the system outputs the walking speed and turning angle. Accordingly, by combining all of these outputs, it is possible to plot the user's path trajectories while traversing over various terrains.

The final chapter (Chapter 7) highlights and discusses the thesis contributions. Then, it answers the scientific questions introduced initially. Finally, the thesis is concluded with some perspectives and future research recommendations.

1.4 Publications

The following chapters are related to peer-reviewed publications. In order to make the logical connections between the chapters in a progressive way, the content is slightly adapted compared to the published articles. The generated publications during this thesis where the authors had significant contributions are listed below.

International conference papers, with peer-review

- Al-dabbagh, Ali HA, and Renaud Ronsse. "A New Terrain Recognition Approach for Predictive Control of Assistive Devices Using Depth Vision." *International Symposium on Wearable Robotics*. Springer, Cham, 2020.
- Al-dabbagh, Ali HA, and Renaud Ronsse. "Using Depth Vision for Terrain Detection during Active Locomotion." *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021.

Journal papers, with peer-review

- Al-dabbagh, Ali HA, and Renaud Ronsse. "A review of terrain detection systems for applications in locomotion assistance." *Robotics and Autonomous Systems* 133 (2020): 103628.
- Al-dabbagh, Ali HA, and Renaud Ronsse. "Depth Vision based Terrain Detection Algorithm during Human Locomotion." *IEEE Transactions on Medical Robotics and Bionics* (2022) (in press).

Terrain detection systems for locomotion assistance: a review

2

Related publication:

- Al-dabbagh, Ali HA, and Renaud Ronsse. "A review of terrain detection systems for applications in locomotion assistance." *Robotics and Autonomous Systems* 133 (2020): 103628.

2.1 Introduction

Literature in legged robotics abounds of examples showing that legs are better than wheels regarding the capacity to handle a large variety of terrains. Similarly, human can navigate through complex terrains like stairs, ramps and rocks. However, humans often have to integrate sensory information from their environment in order to adapt their motions accordingly. Small obstacles like gravels are usually handled through force feedback and short-loop reflexes. In contrast, it is often necessary to visually observe larger terrain variations that would require major adaptations of the

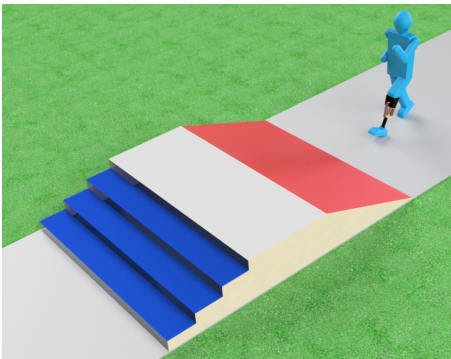


Figure 2.1: Terrain detection is necessary to adapt the controller of a walking assistive device (here, a bionic lower-limb prosthesis) as a function of the task being performed. In this example, the amputee is going to transition between overground walking (gray), slope climbing (red), and stairs descending (blue) [35].

motor plan requiring the recruitment of cortical loops [47]. Without such adaptation, the occurrence of stumbling or falling is ubiquitous due to the large mechanical disturbance that is caused by interacting with an environment featuring a different ground geometry. These closed-loop mechanisms are driven by two types of sensory modalities, namely proprioceptive and exteroceptive sensors. Proprioceptive sensors measure the internal state of the body such as applied forces, joints orientation and limb position. Typical human proprioceptive sensors are the muscle spindles and Golgi tendon organs. In the scope of this work, we will also consider the vestibular system as being part of the proprioceptive pathway, since this sensory modality provides information about internal state of the body, i.e. rotational movements and linear accelerations of the head. Exteroceptive sensors provide information about the external environment such as distance to object and texture [48]. Typical human exteroceptive sensors are vision and touch. The same dichotomy exists in robotics, since robots can be equipped both with proprioceptive (e.g. joint encoders) and exteroceptive sensors (e.g. camera).

The biomechanics of locomotion reveal that leg often produce mechanical energy in various locomotion tasks like walking or stair ascending/descending [5]. Moreover, each of these tasks corresponds to different kinetic and kinematic patterns, even in steady-state [41]. As consequence, active prostheses are necessary to provide lower-limb amputees with the capacity to display healthy gait patterns, and the control of such device should be adaptive to the task being executed (walking, taking stairs, etc.) [49]. In particular, the transition among these tasks should be timely detected in order to adapt the controller (see Fig. 2.1).

Recent papers reviewed locomotion assistance approaches. For instance, control algorithms and assistive strategies used for lower limb prostheses and exoskeletons have been reviewed in [50–52]. More specific reviews focused on intention detection methodologies for lower- and upper-limb prostheses control [53, 54]. Smart walkers (i.e platforms used to assist people with locomotion disabilities) have been reviewed in [55]. Interestingly, all reviews cited above focused on sensors that can be classified as proprioceptive, i.e. providing information about the internal state of the (human) body. More recently, [56] also overviewed research efforts towards the integration of vision sensors in this picture, at least for lower-limb prostheses. Fusing these sensory pathways indeed holds promises for optimizing the detection latency and accuracy in locomotion assistance. Sensory fusion between different modalities is a transversal concept for locomotion intention detection in the context of such devices. Indeed, mechanical (e.g. inertial measurement units, IMUs), electromyographic (EMG), and/or electroencephalographic (EEG) sensors placed on the subject's body capture different aspects of the user's motion, thus reflecting different aspects of his/her locomotion intention. Interestingly, this holds both for the upper- and the lower-limb [21].

This paves the way towards terrain detection for locomotion assistance and, consequently, to the development of detection systems that would *predict* the future user intention rather than *reacting* to changes in behavior being captured by proprioceptive sensors like IMU or EMG. Said differently, perceiving and analyzing the environment around the user would offer to predict potential locomotion intentions several steps in advance.

The objective of the present chapter is to provide an exhaustive review of terrain

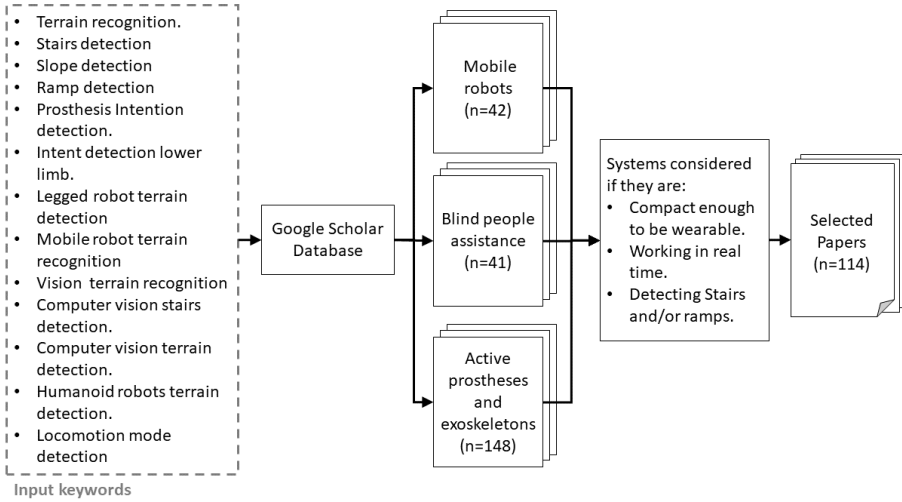


Figure 2.2: Overview of the selection process that was followed to construct the portfolio of this review. First, several keywords were entered in Google Scholar. The corresponding papers were sorted into three categories: those addressing mobile robotic applications (n=42), those providing assistance to blind people (n=41), and those addressing active prostheses and exoskeletons (n=148). For all categories, only developments satisfying further requirements were selected, reaching a total of 114 papers that are analyzed in this review [35].

detection systems for locomotion assistance in the widest perspective. This means that we will also include terrain detection approaches developed for mobile robots pending that they could be applied to assist visually impaired people and disabled walkers (see Fig. 2.2). In sum, three types of terrain detection systems have been pooled in this review:

- Systems implemented for exoskeletons and active prostheses.
- Systems based on vision and ranging sensors to assist visually impaired people.
- Systems developed for terrain detection with mobile robots pending that they are compact enough to be worn by a human and that they can run in real time.

Moreover, this chapter systematically focuses on the detection of simple terrain geometries, mainly corresponding to indoor or urban environments: grounds, slopes, and stairs. Such terrains are indeed faced in everyday life of (disabled) walkers, and illustrate the most critical need for adaptive assistance. Accordingly, the following sections review many sensory systems that were developed for terrain detection. These systems have been classified into two families as illustrated in Fig. 2.3, namely those embedding only proprioceptive sensors (Section 2.2), and those further embedding exteroceptive sensors (Section 2.3). The main features of these systems are then pooled in two comparative tables, whose contents and other aspects are discussed in Section 2.4. Finally, the chapter ends with a conclusion.

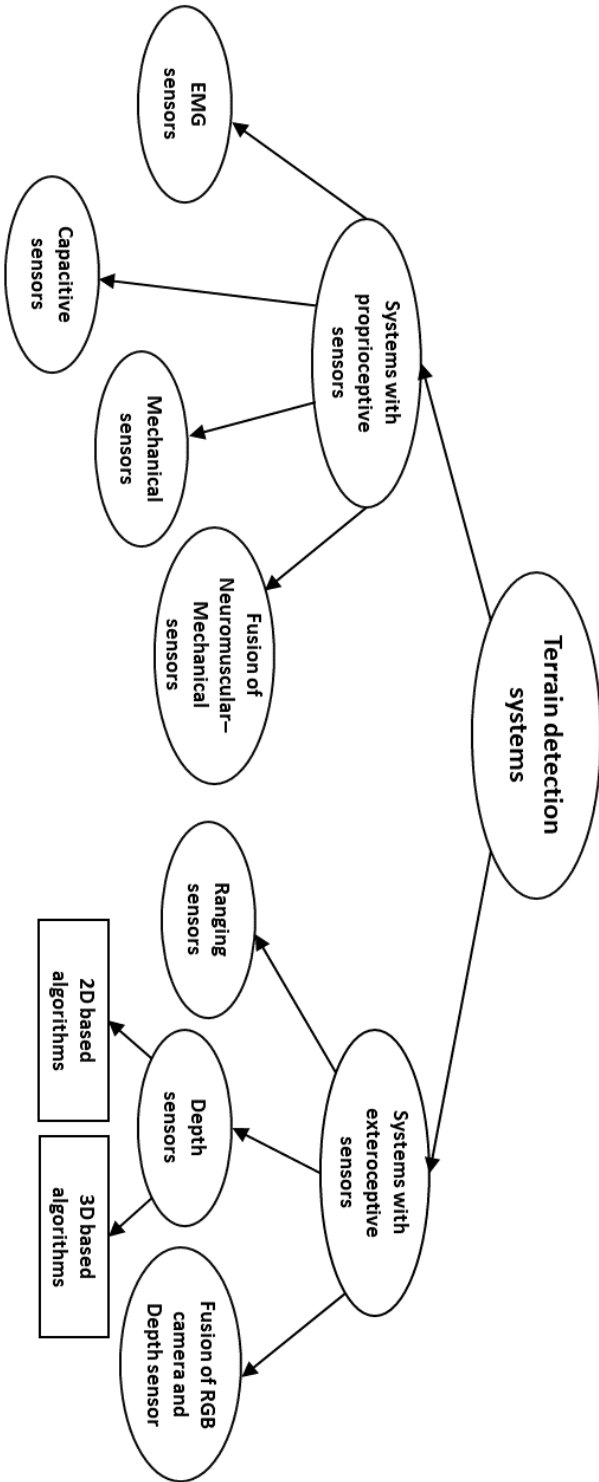


Figure 2.3: This figure shows a classification of typical sensors that are used in terrain detection systems as a function of whether they use proprioceptive sensors only, or also exteroceptive sensors [35].

2.2 Systems with proprioceptive sensors only

These systems use no exteroceptive sensors like vision or ranging sensors. They rather detect terrains indirectly, through a detection of the user's locomotion intention via proprioceptive sensors like EMG, capacitive or mechanical sensors. The first measures the electrical muscle activity, the second determines muscles contraction, and the third measures body kinematics. Typical configurations for placing such sensors on the body are displayed in Fig. 2.4.

A typical processes applied to identify the terrain type with such sensors is shown in Fig. 2.5. Sensory signals are first filtered to remove measurement noise. Then, specific features are extracted from a time window of the filtered signals, such as mean absolute value, standard deviation, minimum and maximum, rate of change, amount of zero crossings, amount of slope sign changes, waveform length, etc. [57–59]. Finally, a classification methodology is applied to identify the terrain type as a function of these features.

Such systems are not suitable to be used for visually impaired people because they typically rely on the user's intact exteroceptive vision for anticipating changes in locomotion mode. Therefore, the literature covering them mostly addresses automatic control adjustment of active prostheses, thus for lower-limb amputees. Accordingly, this section reviews various terrain detection systems utilized for lower-limb prostheses based on EMG, capacitive and mechanical sensors. Note that such terrain detection systems have been already reviewed in [21, 50–55], and consequently, do not represent the main contribution of this chapter. In the following sections, we thus survey and update the main contributions reported in these previous reviews.

2.2.1 EMG sensors

EMG signals generated by leg muscles can be used to identify the terrain type which the subject is moving on. EMGs are electrical signals that are sent to muscles to trigger their contraction [60], and they can be measured through sensors placed on the skin above the muscle. If the person is moving on different types of terrains (slopes, stairs and level ground), different EMG patterns will be observed [61]. Data processing techniques are applied to extract features from these signals in order to identify the terrain category, and enable the prosthetic controller to select the corresponding

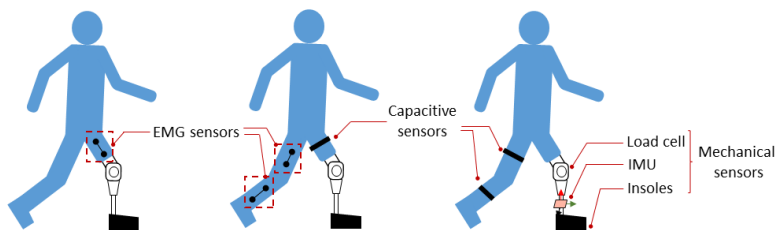


Figure 2.4: Representative configurations of sensors for terrain and locomotion intention detection with a lower-limb amputee wearing a prosthesis. It shows the possible distribution of proprioceptive sensors measuring muscular activity (EMG), muscle contraction (capacitive), or the leg kinematics (Mechanical) [35].

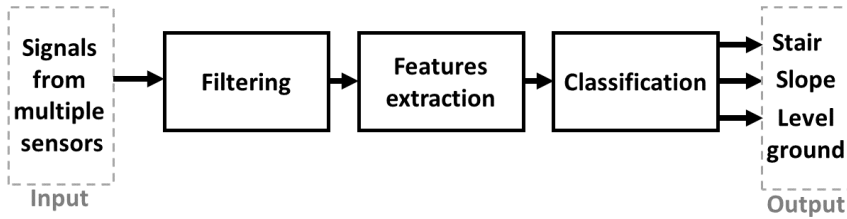


Figure 2.5: Processes applied to detect various terrains based on EMG, mechanical or capacitive sensors [35].

control mode [62–64]. Electrical signal from the muscle is typically detectable 100ms prior to the actual movement [60]. However, terrain identification requires filtering, feature extraction and classification, and thus causes computational delay [21].

Various methodologies have been developed to incorporate locomotion intention and/or terrain detection from EMG signals in the closed-loop controller of active prostheses. For instance, a finite state controller can be used to modulate the impedance and output power of the prosthesis as a function of the EMG-based classification [63, 65, 66]. More precisely, [66] implemented two finite state controllers to induce biomimetic movement patterns for level-ground and stair-descent gaits. In that contribution, the subject locomotion intent was inferred through classifying the EMG features with a neural network, and supervise the transition between finite state controllers.

Another approach consists in classifying the set of measured EMG features with classifiers such as Support Vector Machine (SVM) or Linear Discriminant Analysis (LDA) [67, 68]. In particular, a recent study tested applying SVM, LDA and Neural Network (NN) classifiers on single channel EMG for locomotion mode recognition [69]. The study also incorporated a feature selection algorithm which improved the performance of all the classifiers tested. The classifiers reached above 96% accuracy with slight differences from one to another [69]. Globally, there is a trend to promote LDA as the most appropriate classifier for EMG pattern recognition since it has similar accuracy as more complex classifiers, but it is computationally more efficient [68, 70–72].

2.2.2 Capacitive sensors

Capacitive sensing relies on measuring the natural electrical capacitance of the human body. It has many applications, including human machine interfaces [73]. This technology has also been used to detect different locomotion modes by measuring changes in the leg configuration during locomotive movements. More precisely, this requires to measure the impedance between transmitter and receiver arrays of electrodes placed on the leg so that the leg dielectric capacitance captures morphological changes in soft tissues like muscles. This produces different output impedance profiles for different locomotion modes [74].

Similar to the EMG sensors, various classifiers can be used to identify locomotion modes from several capacitive sensors placed over the leg. Due to its low computational load and good performance, LDA has been used for human gait identifica-

tion based on wearable capacitive sensors placed in direct contact with leg skin [75]. Moreover, LDA, QDA (Quadratic Discriminant Analysis) and LR (logistic regression) classifiers have been tested in [74] with the same hardware. LDA displayed similar performance as LR, and smaller classification error than QDA.

Non-contact capacitive sensing is an alternative approach of capacitive sensing. In such systems, copper wire mesh pieces are fixed on the outer surface of the prosthetic socket. Accordingly, the human body and the meshes form the capacitor electrodes, while the plastic material between them form the dielectric. During human movements, the physical distance between two electrodes changes, and this will result in different output voltage patterns across various locomotion modes [59].

In a non-contact capacitive sensing system, the QDA classifier outperformed both LDA and GMM in average locomotion mode identification accuracy by a small margin [59]. In contrast, SVM showed slightly higher accuracy than QDA regarding terrain recognition with a non-contact capacitive sensing based system [76].

2.2.3 Mechanical sensors

Mechanical sensors directly measure the human body kinematics (e.g. with IMUs) and interaction forces between the body and its environment (such as with instrumented insoles or load cells). Interaction force sensors lie therefore at the boundary between proprioception and exteroception since they measure a signal at the interface between the user and its environment. Such sensors reflect the current state of the user's segment, and/or of the prosthesis [77]. Various approaches have been developed to achieve terrain identification with this series of sensors.

Similarly to EMG signal classification, LDA has been widely used to classify features extracted from mechanical sensors data. For instance, merged signals from three IMUs and two foot pressure insoles have been used to achieve locomotion (transition) detection with a two-level LDA [78]. In another contributions, [79, 80] developed two LDA classifiers mixing several mechanical sensors. One classifier focused on data collected at heel contact and the other at toe-off, in order to detect transitions between different locomotion modes at least twice per locomotion cycle. LDA is sometimes further combined with other classifiers. For instance, combined LDA with a decision tree analysis (DTA) classifier are used to identify locomotion mode using only wireless foot insoles [81]. In [82], LDA has been used as a first stage classifier (followed by SVM) for feature extraction of IMU data mounted above the ankle to identify human gait kinematics. Interestingly, it displayed higher accuracy than using SVM alone. In contrast, a Phase Variable classifier (PV) presented in [83] showed higher accuracy than LDA when it is trained with nonsubject-specific data, while LDA outperformed PV with subject-specific training data. Another type of Gaussian distribution-based classifier (QDA), has been used in a cascaded classification algorithm [84]. More precisely, it was developed for intent recognition based on data from IMUs and load cell on the prosthesis. A first level classifier discriminates between standing and walking, a second level identifies the main gait phases (namely swing and stance), while the third level recognizes steady-state locomotion modes such as level ground walking, stair ascending, descending, ramp ascending and descending and the transitions between those modes.

Other families of classifiers have been further tested for intention detection and

terrain identification. For instance, [77, 85] used Gaussian mixture models (GMMs) to achieve real-time gait mode intent recognition. More complex classification approaches such as neural networks have been used to identify different gaits based on force sensors in the prosthesis socket, or IMU data [86–88]. Recently, a study compared a LDA classifier versus heuristic prediction algorithm using thresholds on estimated prosthetic ankle joint translations and ground slope [89]. Internal prosthesis sensory data (with mechanical sensors only) were fed to be classified by each method. The study showed that the heuristic prediction algorithm has slightly lower prediction error than the LDA classifier (2.8% vs. 3.4%). Interestingly, the heuristic prediction took only five features while the LDA required 45 features to reach such a low prediction error.

In order to enhance intent recognition performance, and make a more accurate prediction, many researchers added prior information using e.g. dynamic Bayesian networks (DBN) or hidden Markov models (HMM) [90–92]. For instance, [93] introduced a user independent intent recognition system based on classifying the data from 13 mechanical sensors (encoders, potentiometers, load cells and IMUs) directly integrated into a powered knee and ankle prosthesis. They tested various classification approaches and the history information included prior beliefs about locomotion history through DBN. Finally, it is suggested that fusion between capacitive and mechanical sensors could provide a good alternative to systems relying on the fusion between EMG and mechanical sensors [76], although further research is needed to support this claim.

Finally, recent studies paved the way towards combining IMU sensors only and advanced classification approaches. For instance, an approach presented by [94] converts the IMUs data into grayscale images that can then be classified into different modes of locomotion by a Convolutional Neural Network (CNN). The system uses 3-IMUs mounted on the thigh, shank, and ankle of the sound leg and achieved about 89% accuracy. Another contribution reached higher accuracy (about 98%) by using a single 9-axes IMU sensor placed on the sound foot [95]. The system tracks the sensor trajectory which reflects foot trajectory, and it is used to derive the inclination grade of the ground and locomotion mode recognition. The calculated trajectories are then analyzed for predicting the mode of locomotion.

2.2.4 Fusion of Neuromuscular and Mechanical sensors

Sensor fusion is combining data from multiple sensors in order to get higher accuracy for detection or classification. Human and animal brains routinely accomplish sensory fusion in order to optimally exploit information provided from their multiple senses such as audition and vision [96, 97]. Sensory fusion has also been applied in many technological fields such as medical and mobile robots [96–99]. For instance, data fusion has been applied in [100] to identify human daily activities based on multiple IMU sensors. In this section, we will specifically review the methods and results related to the fusion between EMG and mechanical sensors, i.e. the most used modalities for locomotion intention detection and terrain identification.

Locomotion features from mechanical sensors data and EMG signals have been extracted and concatenated in a single feature vector to be classified with various approaches [101]. One relies on using non-linear or linear classifiers such as SVM

or LDA [42]. In general, SVM showed better classification accuracy and prediction as compared to LDA. More precisely, SVM predicted transitions between locomotion task modes before the critical event, and earlier than linear classifiers [42]. However, integrating prior information to LDA classifier resulted in improved accuracy and prediction time than using LDA without integrating this prior knowledge. These information was integrated through prior probabilities adjustment of discriminant functions of LDA [102].

In contrast, researchers in the field of neuromechanical sensors fusion considered alternative approaches focusing on hardware enhancement. For instance, using an FPGA as the computational engine with LDA classifier for fast decoding and recognition resulted in better prediction time with the same accuracy [103]. Another hardware breakthrough consisted in using a Graphics Processing Unit (GPU) instead of Central Processing Unit (CPU). This provided faster response time for transitions with an SVM classifier. The system was evaluated using virtual reality with an avatar emulating intended movements [104]. Alternatively, a mobile CPU based system was presented in [105] targeting minimization of power consumption with a real-time SVM classifier.

In general, systems relying on fused neuromuscular and mechanical sensors outperformed other approaches based on one kind of information source only [42, 102, 104]. However, in order to minimize the computational time and complexity, it is interesting to know which sources of information are the most valuable for achieving the highest accuracy among all mechanical and EMG sensors that could be used to instrument the human (lower-body). In [106], a source selection algorithm called minimum-redundancy-maximum-relevance (mRMR) was used for this purpose. Interestingly, the system was able to recognize the subject's locomotion intent with an accuracy of more than 95% when data sources were reduced from 26 to 10, despite a decrease in computational cost by a factor of about 1.9. Finally, it is suggested that accurate terrain detection requires the prosthesis to be equipped with at least four surface EMG sensors, a 6-DOF load cell mounted and one IMU [106].

2.2.5 Summary

In summary, representative systems using only proprioceptive sensors are listed in Table 2.1. It reports the terrains that can be detected by each approach, the corresponding sensors, and the developed classification algorithms. Note that all these methods have been tested both in indoor and outdoor environments. More quantitatively, the table also reports the detection accuracy and prediction time, when available. This prediction time T_{pred} is a metric assessing the performance. We defined it as the difference between the critical time t_c , i.e. the time when the locomotion mode changes or should change due to terrain variation, and the time when this change is actually detected, i.e. the detection time t_d [107]:

$$T_{pred} = t_c - t_d \quad (2.1)$$

If $T_{pred} > 0$, the corresponding detection system can thus *anticipate* (or predict) the change of locomotion mode; while if $T_{pred} < 0$, the corresponding system would only *react* to a change of locomotion mode with some latency. According to Fig. 2.6,

Table 2.1: Overview of representative systems with proprioceptive sensors only. The last two columns report the average prediction time and detection accuracy — when available — for the representative task of transitioning from overground walking to stair climbing. $0s < T_{pred} < 1s$ captures that all these methods detected the transition before it actually happened, but in a time window that is smaller than 1s.

References	Targeted rains	Ter-	Sensor	Algorithm	$T_{pred}[ms]$	detection accuracy[%]
[68]	Stairs, ramps		EMG electrodes	LDA classifier	\	97
[42]	Stairs, ramps		EMG electrodes	SVM classifier	254 ± 132	\
[78]	Stairs, ramps		Mechanical sensors	Two level LDA classifier	222 ± 14	99.71
[84]	Stairs, ramps		Mechanical sensors	Cascade classification (QDA based)	64 ± 133	93.21
[95]	Stairs, ramps		Single IMU sensor	IMU trajectory tracking	202	98
[42, 102]	Stairs, ramps		Neuro-mechanical fusion	LDA classifier	340 ± 136	96
[102]	Stairs, ramps		Neuro-mechanical fusion	SVM classifier	420 ± 175	99
[103]	Stairs		Neuro-mechanical fusion	LDA with prior knowledge	415 ± 138	98
[106]	Stairs, ramps		Neuro-mechanical fusion	LDA based classifiers with FPGA as computing engine	549.83 ± 139.2	99.31
				All sources with SVM classifier	120	98
[74]	Stairs		Capacitive sensing (C-Sens)	Source selection algorithm with SVM	119.3	95
[59]	Stairs		Contactless C-Sens	LDA classifier	\	93
[76]	Stairs		Contactless C-Sens	QDA classifier	\	94.8
				SVM classifier	\	95.8

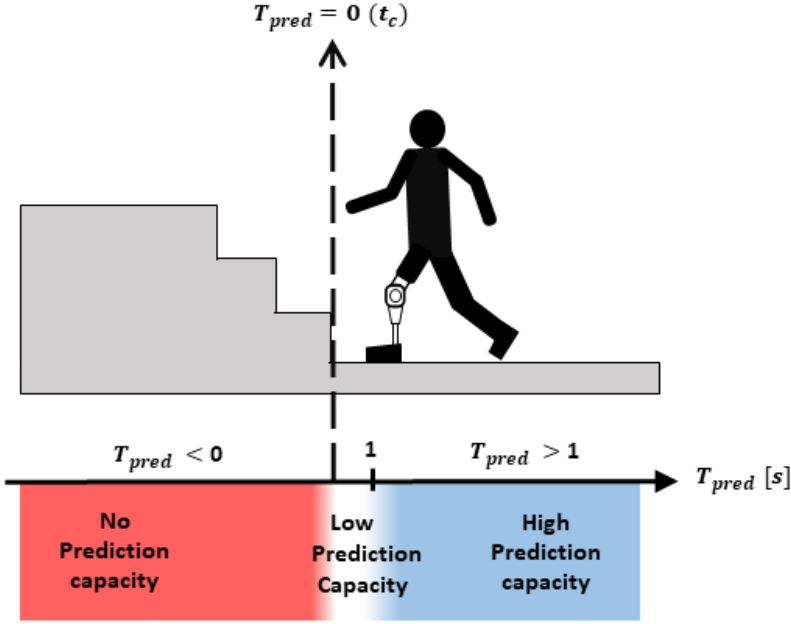


Figure 2.6: This figure highlights that prediction time (T_{pred}) is a critical metric to assess the performance of a terrain identification system. Systems with high prediction capacity ($T_{pred} > 1$ s) have a detection time t_d that occurs more than one second before the critical time t_c where a locomotion event is taking place (here, a transition between walking and stairs ascending). Systems with low prediction capacity ($0s > T_{pred} > 1s$) have a prediction time of the same order than one footstep. Finally, systems with no prediction capacity detect the transition after the critical event took place, i.e. the prediction time is negative ($T_{pred} < 0s$) [35].

we further used this prediction time as a metric for classifying the identified systems into two categories: those having no to low prediction capacities, i.e. with $T_{pred} < 1s$, vs. those having prediction capacities larger than 1s. This threshold of one second – thus approximately corresponding to one step – was considered to be the minimum detection time being necessary for incorporating an anticipation of locomotion change within the device high-level controller. Note that several papers report different levels of accuracy and prediction time, as a function of the task transition (i.e. ground to stairs, stairs to ground, etc.). For the sake of making fair comparisons, only data corresponding to a transition from level ground to stairs ascending are reported, since this is likely the most frequent and most representative transition requiring a macroscopic adaptation of the device controller [93]. Therefore, all systems reported in Table 2.1 can thus be classified as having no to low prediction capacity, since T_{pred} is systematically lower than 1s.

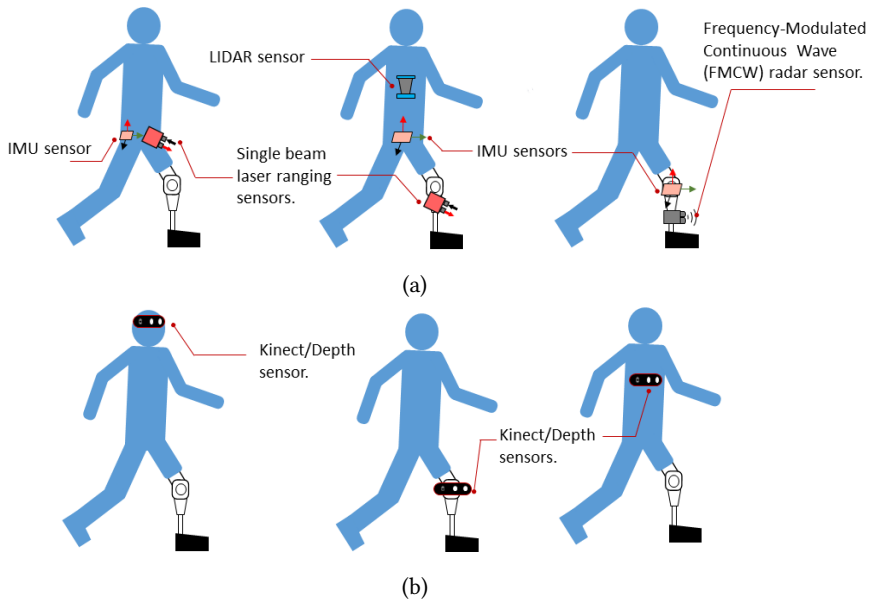


Figure 2.7: Representative configurations of exteroceptive sensors for terrain and locomotion intention detection with a lower-limb amputee wearing a prosthesis. Panel (a) highlights exteroceptive ranging sensors, that can be either laser-based or FMCW, and that can be placed on several body parts. Similarly, panel (b) highlights possible placements of exteroceptive vision sensors (depth). This also represents possible sensor configurations to assist visually impaired people [35].

2.3 Systems with at least one exteroceptive sensor

Next to the systems reported in the previous section, several contributions further focused on equipping locomotion assistive devices with large prediction capacities, typically corresponding to a few steps ahead of time. These systems rely on exteroceptive sensors, i.e. sensors receiving stimuli originating from the environment, outside of the body. They typically deploy various kinds of ranging or vision sensors observing the terrain in front of the subject. Their large prediction capability relies on detecting the terrain type before the subject steps into it. One traversal challenge with these sensors that they move with the wearer's body. Therefore, an accurate and dynamic estimation of the sensor orientation is necessary to transform the obtained data [108]. Consequently, many implementations of these systems integrate IMUs. Importantly, a vast literature of terrain detection systems with vision or ranging sensors has been developed for mobile robots, towards autonomous navigation and obstacle avoidance. Interestingly, these approaches could be transposed to locomotion assistance with similar hardware and algorithms. In the following sections, we will pool several terrain detection strategies that were developed either directly for human locomotion assistance (including visually impaired people and amputees) or for mobile robot navigation, pending that adaptation to locomotion assistance would be feasible.

2.3.1 Ranging Sensors

These sensors can be classified as a function of whether they emit a single or multiple beams. Actually, many approaches introduced for terrain detection used single beam sensors. For instance, [109] developed a system to identify level ground, up and down stairs with a single ultrasonic sensor. Data collected from this sensor over time—while the subject was moving—has been classified with SVM. However, the system did not achieve an average classification accuracy higher than 70% [109]. Another development consisted in mounting a single beam laser sensor and an IMU on one side of the subject's waist (see Fig. 2.7(a) for illustration). Authors showed that it is possible to detect various terrain types with classification or decision tree [110–112]. In particular, the classification tree used the terrain height as primary feature for recognition [113]. Although authors reported a classification accuracy as high as 98%, this system achieved prediction time of about 700ms. In such system, the user should keep his/her hands away from the waist to avoid covering the sensor and preventing detection [43]. A possible solution is to fix the laser sensor on the subject shank (or on the prosthesis), but this configuration produces noisy measurements, and did not show obvious differences in pattern among terrains that were tested [111]. However, using a wearable Frequency Modulated Continuous-Wave (FMCW) radar attached to a lower limb prosthesis succeeded in stairs detection [41]. The radar position is pictured in Fig. 2.7(a). It scans the objects in front of the prosthesis in the sagittal plane during the rotational movement of the tibia. Distances to the detected objects are further measured during the stance phase. This resulted in multiple distances that must then be transformed to the global frame. To get rid of clutter noise, the Cell-Averaging CA-CFAR algorithm presented in [114] was applied. 2D radar scans were converted into a binary image, and the reflections of each scan on objects were casted as ellipse shaped regions [115]. On the other hand, it is important to point that single range sensor do not allow to detect some important features of the environment, like stairs width and height. Also, when the subject is standing or turning, the recognition process is disabled because it requires the subject to move forward in order to scan the terrain [43]. A recent new method used multiple ranging sensors placed on the shank of the user's leg. When it becomes perpendicular to the ground during the stance phase, ranging sensors are triggered to observe the terrain type in front of the user. The system reached 99% of classification accuracy [116].

With multi-beam ranging sensor, it is possible to robustly detect a richer repertoire of terrain features. A first approach presented a step detection system for the visually impaired in [117]. It used LIDAR sensor mounted on the subject's chest, and provides distance estimates in the vertical plane. Neighboring points are connected to form multiple segments, further classified into vertical and horizontal. Stairs detection is done by comparing each segment length with a threshold [117].

Another stairs and steps detection system introduced by [118] also used a single LIDAR. The sensor beam is divided into three beams (two vertical and one horizontal) through reflective mirrors to ease steps detection. This system is mounted on a wheelchair for disabled people; and would likely be too bulky to be used as a wearable sensor.

Estimating the 3D structure of the environment leads to more robust terrain detection [119]. This can be achieved with 2D-LIDAR, 3D-LIDAR, or Depth sensors. There are many techniques used to obtain a 3D estimation of the environment with

2D-LIDAR sensors through adding another LIDAR or using an arm to scan the environment. However, such adaptations are mostly applied with wheel chairs or mobile robots [120–123], and adapting this to a wearable system would again be infringed owing to the large size of hardware components and delay of scanning, while using 3D-LIDAR such as in [124, 125] is also costly and bulky. An alternative solution is to use a depth sensor as discussed in the next section.

2.3.2 Depth Sensors

Depth sensors have been widely used as wearable terrain detection systems for locomotion assistance mounted on mobile robot for navigation purposes. This is because they are low-cost, compact, and robust to various textures and lightning conditions. Moreover, they provide accurate mapping of the environment leading to robust terrain type recognition [119].

Depth sensors are said to be either passive or active. For the first kind, two aligned RGB cameras are used, and range is computed by triangulation between matching pixels in both images [126]. In contrast, an active depth sensor measures the range either by detecting the light wave phase shift between the emitted and received light (the so-called time of flight camera, TOF) [127], or by determining the disparity between a projected infrared light pattern and the measured one [128]. This is the working principle of the famous Microsoft Kinect. In comparison, active depth sensors can work in low illumination or even dark places, while stereo matching may fail in low contrast environments [129]. Some active sensors are affected by the sunlight [130].

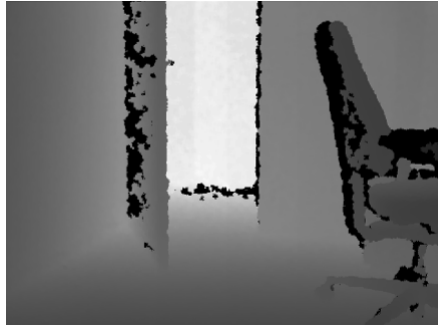
Targeting locomotion assistance, such a depth sensor can be typically placed on three different parts of the human body, as illustrated in Fig. 2.7(b):

- On the head: mounting the depth camera on the head is typically used for assisting people with visual impairment. Two configurations are found in the literature. The first consists in mounting two cameras on specific glasses, and through stereo matching, depth data can be reconstructed [131]. The second requires to fix an active depth sensor or stereo cameras on a helmet worn by the user [132–134].
- On the leg: this configuration is mainly used with lower limb amputees. The depth camera can be mounted on either the sound or the prosthetic leg, and either above or below the knee [43, 135, 136].
- On the chest: this approach is appealing both for visually impaired individuals and lower limb amputees. Moreover, it tends to provide more stable images than concurrent configurations [119, 137].

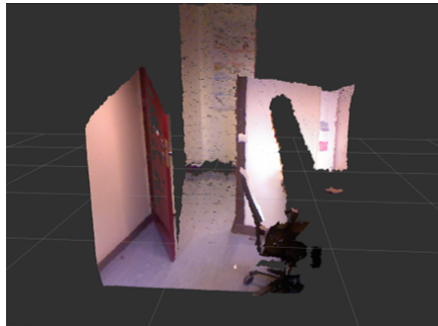
Both passive and active depth sensors provide range information as a gray scale depth image or point cloud data. In a depth image, each pixel corresponds to the same image pixel, and the gray level corresponds to the distance between this point and the sensor [138]. With the point cloud approach, the environment is represented by a set of 3D points in space and the coordinates of each point correspond its 3D location in the real world, centered on the camera. An example of depth image and point cloud for the same scene is shown in Fig. 2.8. There are two types of methodologies



(a)



(b)



(c)

Figure 2.8: (a) original RGB image, (b) depth image, and (c) corresponding point cloud [35]. The images are captured with Kinect sensor.

to process information acquired from depth sensors. The first method consists in applying 2D image processing approaches on depth image, while the second method consists of applying 3D vision algorithms on point cloud or depth data. In the next subsections, we will explore both of these approaches for terrain detection.

2.3.2.a 2D-based algorithms

These approaches are applied on depth image, and with such methods, it is possible to detect terrains having a sharp variations in height such as stairs, steps, and holes. This process of terrain detection from a depth image is sketched in Fig. 2.9.

Firstly, edges are detected with a corresponding algorithm such as Sobel or Canny. Details retention is reported to be better with the later than the former [139]. However, since stairs or step edges are easy to detect in depth images, Sobel is often preferred for its cheapest computational cost [140–144] and less contributions reported the use of Canny [145, 146]. The second step consists in extracting the horizontal lines in the image, for instance by using the Hough transform [147, 148]. Finally, the detected lines are filtered in order to discriminate convex and concave edges corresponding to steps as shown in Fig. 2.10. This filtering is done through applying various constrains. Such as:

- A threshold on the tilting angle of the detected lines, in order to filter out the vertical and non-horizontal edges [140, 141, 143]. It is, however, possible to omit this step, and directly isolating the horizontal lines with a Hough transform [142].
- Lines whose length is below a certain threshold are removed, in order to keep only those that can be part of steps. [140–143]
- The vertical distance between two consecutive lines should be within a certain range, in order to remove the detected lines being either very close to each other (multiple detection of the same edge) or far away and thus do not belonging to the same object [143].
- If the horizontal distance between two lines is smaller than a specified threshold, then these lines are assumed to belong to the same edge that could be e.g. obstructed by an obstacle [140, 141].
- The depth difference between two adjacent convex lines is typically used to compute the size of footstools. If it is smaller than a specific value, then the detected line is assumed to be not belonging to a staircase [143].
- Finally, the number of parallel lines is sometimes used to reject stair case candidates if it is not composed of enough edges [146]. Also, this number can be used to distinguish between single steps (or curbs) and stairs.

After detecting the stair line candidates, an extra verification can be applied. For instance, a plane is fitted through the parallel lines, and if its slope is out of a specific range, the staircase candidate is rejected [146]. An alternative approach is to fit a line through all the candidates, and obtain a single dimensional feature which contains the depth information as a function of the coordinate along the

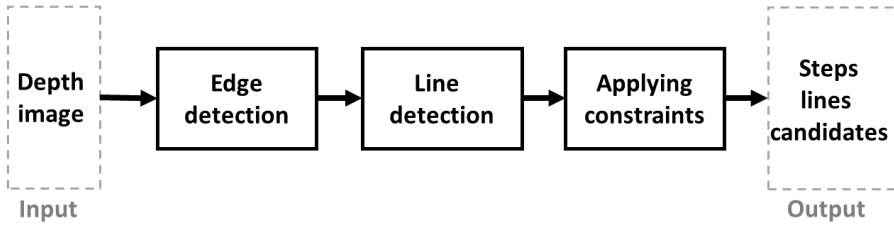


Figure 2.9: Stair and step detection process with depth image

— convex
— concave

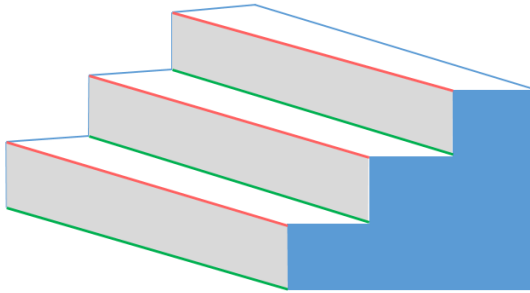


Figure 2.10: convex(red) and concave(green) edges of a typical staircase [35].

forward axis of the depth image. Then, the feature is classified using SVM or other classifiers [140, 143].

Other researchers developed an algorithm to identify human gaits based on the difference between two depth images. This differential image is generated by subtracting multiple consecutive depth frames from each other. The resulted image is classified using SVM in order to identify the human gait during different activities such as stair climbing, descending, and level-ground walking [135, 136].

2.3.2.b 3D-based algorithms

These methods are applied on point cloud or depth data. It is possible to detect various terrain types such as slopes, steps, holes, and uneven ground based on such scheme, and there are several techniques achieving this classification. The overall process of terrain detection based on point cloud data is shown in Fig. 2.11. First of all, the point cloud is passed to multiple processes before feeding it to the main algorithm for terrains recognition. These filters are typically applying low-pass filtering, down sampling, transformation, normal estimation and plane detection. After that, the computed planes are fed to a detection algorithm identifying the terrains in front of the sensor. Furthermore, other computations can be performed in parallel to the plane detection, as illustrated in Fig. 2.11. In the following sections, we will review these commonly used filters and algorithms.

Pre-processing: This step aims at improving the signal to noise ratio and reducing the computational load. This is achieved by using various filters such as outlier

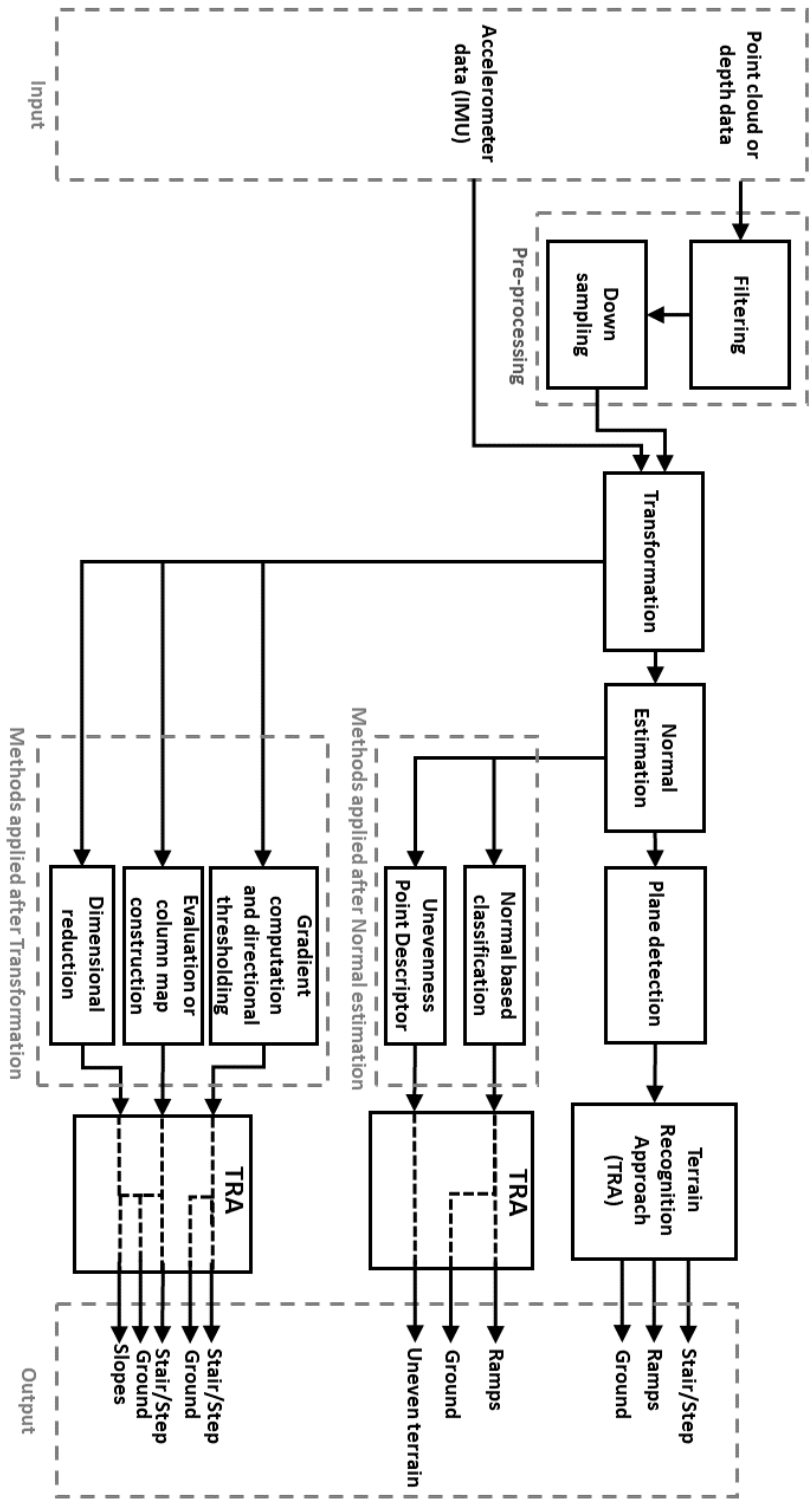


Figure 2.11: Overall process of terrain detection systems based on point cloud and accelerometer/IMU data. Typical process starts with point cloud filtering and down sampling. Then in the transformation step, the point cloud is mapped to the world coordinates. After that, the point cloud normals are estimated, and a plane detection algorithm is applied. Finally the terrains are detected and modeled. However, there are various other methodologies that can be applied after transformation or normal estimation processes which lead to terrains detection and modeling [35].

removal and averaging [149,150], although this comes with the cost of decreasing the frame rate [151]. Computational savings are obtained by down-sampling. Indeed, the high density of point cloud data causes processing delay, and not all the points bring a significant part of new information. The point cloud resolution is down sized by using algorithms such as 3D voxel grid filter [152], Narrow Down Filtering (NDF) [153], or by averaging with a sliding 2×2 block [154]. To further reduce the computational cost, a region of interest (ROI) can be specified in order to eliminate data lying out of this region. For instance, [155] specified a 150 cm wide ROI in front of the user, so that the points lying out of this range were not treated.

Transformation: Point cloud data acquired from depth sensor are provided in a coordinate frame being centered on the camera, and this causes issues for proper terrain detection. For instance, if the camera points downwards, level ground will be detected as a ramp [43]. A solution is to transform the reference coordinate system to the ground or world coordinates [137,156]. In general, the depth sensor orientation is acquired either from an IMU embedded within the camera (such as with the Kinect [156]), or from an external IMU attached nearby [43]. This transformation can also be applied on the depth data [156].

Normal estimation: For each point, a normal vector is estimated from its neighboring points [157]. Various approaches were developed for this computation, balancing the classical trade-off between the accuracy and computational cost [158]. One simple approach consists of computing the cross product of two vectors being tangent to the surface, giving thus the vector being normal to it at a specific point [133,157,159]. Another methodology involves estimating the normals based on Principal Component Analysis (PCA) [119,137,151]. This approach is efficient and has been used by many researchers [149,160,161]. It computes the eigenvalue and eigenvector of a covariance matrix created from direct neighboring points. The normal direction is estimated as the eigenvector having the smallest eigenvalue [137]. After this normal estimation, point cloud data can be segmented into multiple groups based on their orientation. This brings the identification of relevant features like ground plane, ceiling, walls or slopes [151]. Also, it is possible to filter out some irrelevant groups of points away [132,154,155].

Plane detection: This step consists of fitting a 3D plane model on a set of point cloud data, and again this can be performed using various methodologies. A common approach relies on the RANDOM SAMPLE Consensus (RANSAC) algorithm presented by Fischer and Bolles [162]. Basically, it repeats two iterative steps on a given point cloud. First, a hypothesis is made and then it is verified [163]. This algorithm has been frequently applied for human locomotion assistance [132,155,164,165] or mobile robot navigation [153]. The output of this algorithm is the set of detected planes and their normals. A concurrent methodology for plane estimation is regional growing. It starts from a point with minimum curvature called *seed*, and expands the region to include neighboring points having quasi-parallel normals and similar curvature [119]. When the region cannot be expanded anymore, a new seed is chosen, and the process is repeated until a pre-specified threshold is reached. This algorithm outputs a set of points (clusters) belonging to the same smooth surface [152]. Some people argued that the regional growing algorithm has a better physical relevance since the planes are detected from a closed region depending on single element (seed) rather than from set of uncorrelated points distributed in the scene [119,137].

Terrain detection algorithms: Various methods can be applied after plane detection to recognize different types of terrains. For example, a ramp lying in front of the subject and its tilting angle can be identified from the detected plane normal. This can be implemented with regional growing and RANSAC to capture an inclined plane over the cluster having a normal tilted with respect to the ground plane [166]. Also relying on normal filtering, a more complicated framework has been developed for humanoid robots, based on the detection of the geometric primitives in the environment. What each primitive can offer to the individuals (called "affordances" by the authors) is obtained based on its normal vector and area. For example, the object affords support if it is in the same orientation as the ground and its area is larger than a pre-defined threshold. A similar approach can be used for stairs, slope, and ground detection [27].

Alternatively, a common approach consists of identifying the step candidates from their height and orientation [132, 154, 155]. For instance, the tread of each step has the same orientation as the ground, although it is located at a different height. Each candidate (starting from the lowest) can thus be examined to be either classified as a step or an obstacle (a table for instance) [119]. On the other hand, if the height is negative, the terrain in front of the user is likely either a descending stair or a hole, which can be again tentatively discriminated by using predefined thresholds. In another approach, the stairs convex edges are identified from the detected planes, by applying a Concave Hull algorithm followed by normal based filtering. This methodology is applied to detect single or multiple steps in both ascending and descending stairs [153].

After that, a molding algorithm can be applied to compute the number of steps and their features such as tread width, length and height of the riser [137]. Again, different approaches have been tested, such as PCA or Preemptive RANSAC [132, 137].

Most of the approaches for terrain detection with depth sensors are usually derived from one of the processes described previously. The following sections review some of the most common among them. According to Fig. 2.11, there are two types of methodologies: methods used after the transformation step, and those being applied after the normal estimation process.

Methods applied after transformation: Based on depth data, an algorithm consists of computing the normal gradient and applying directional thresholding on the depth data in the three main spacial directions [156]. This algorithm extracts 3D information from 2D depth image. It is usually followed by a stair detection and modeling algorithm for estimating the number of steps [157]. This approach has already been applied to assist visually impaired people [133]. However, based on point cloud, identifying steps and slopes can be done by applying point cloud dimensional reduction after the transformation process [167]. Then, the resulting single dimensional feature is classified with a neural network algorithm, and stairs can be modeled using 2D RANSAC [43, 44]. An alternative algorithm for steps and slopes detection relies on the so-called column map estimation [168, 169]. These maps are constructed through 2D-grid located on the ground plane. Then, the 3D points are mapped to individual cubes located on the grid. Points lying outside the cubes are ignored. The evaluation histogram map is then computed through a voting scheme. This approach can be used to detect steps, slopes and holes [169, 170]. It has

been developed for legged mobile robots [170, 171], although nothing precludes its adaptation to for human locomotion assistance.

Methods applied after normal estimation: Many approaches can also be applied right after the normal estimation process. A simple algorithm detects ground and slopes by classifying point cloud data according to their normal [151]. Then, the remaining point cloud data are used to detect and model stair-cases with a histogram based method [151].

Finally, non-traversable or uneven grounds can be detected using Unevenness Point Descriptor (UPD), again using the normal vectors estimation [149]. UPD can also detect terrain inclinations [160]. This algorithm has been developed mainly for mobile robots to avoid highly irregular grounds [161]. Again, nothing would prevent the adaptation of this framework for human locomotion assistance. Indeed, detecting uneven ground can be used to warn the visually impaired walking towards an hazardous zone. Similarly, it could be applied to adjust the control parameters of a lower limb prosthesis to provide assistance adapted to uneven surfaces, and thus mitigate the risk of falling [12].

2.3.3 Fusion of RGB and Depth Sensors

An RGB camera is a cheap sensor, compact in size, and covers wide angle and long range. Literature showed that it was possible to detect various terrain types such as stairs [172], slopes [173] or ground with a single RGB camera [174]. Indeed, the process of stair detection in a RGB image is similar to the one used for depth images, pending that the image is converted to gray scale and denoised with appropriate filters [175]. Another methodology is to train a deep neural network with RGB images for human locomotion mode classification. Such systems achieved high accuracy in identifying stairs ascending, stairs descending and level ground locomotion modes [45, 46]. However, more advanced features like dimensions or distances can hardly be estimated from such an image. Also, such sensors are useless in dark environments [176].

Consequently, fusion of depth with RGB images has been proposed to get better detection results. The depth sensor being affected by the sunlight, thus the RGB camera is used for stairs detection in outdoor scenarios [142]. Another approach used the RGB camera for stair edges detection, and the depth image for identification of the slopes (ascending or descending) [143]. Similarly, a stair detection system for visually impaired people assistance combined an RGB camera for stair candidate detection with the depth information acquired from stereo images to exclude false positive candidates based on the ground plane detection [131]. This fusion is also useful for distinguishing between pedestrian cross walks and stairs to assist visually impaired people. Indeed, both generate consecutive horizontal lines in the RGB image but can be discriminated by using depth information. [140].

2.3.4 Summary

Similarly to what was done for systems with proprioceptive sensors only, a list of representative systems embedding exteroceptive sensors is provided in Table 2.2. Moreover, the table reports the ability of the systems to work in indoor or outdoor

environments based on the sensor limitation or on the tested environments mentioned in the referenced paper. The table further reports the maximum detection distance.

To have a consistent comparison between systems with proprioceptive and exteroceptive sensors, the prediction time T_{pred} is also reported in Table 2.2. However, T_{pred} is not reported in most of the papers developing systems with vision or ranging sensors. They rather tend to report the maximum distance from which the system can detect the terrains. Consequently, we computed an approximation of T_{pred} based on the ratio between the maximum detection distance, and a typical human walking speed.

According to [177, 178], the comfortable walking speed of transfemoral amputees is 3.6 km/h, and it is 4.53 km/h for transtibial amputees [177]. Therefore, we took 4 km/h as an average between both values. In some papers, the authors didn't mention the maximum detection range of their systems; therefore, we considered a typical active range of the sensor that was used, as reported in its datasheet.

2.4 Discussion

In the previous sections, terrain detection systems for human locomotion assistance are classified into two categories depending on whether they rely on proprioceptive sensors only or on exteroceptive sensors as well. The following paragraphs review the most important trends that are emerging from this extensive reporting, regarding both hardware and software. We will start with discussing the main trends appearing within each category. After that, a comparison between both categories is reported, leveraging several important features.

For systems with proprioceptive sensors only, Table 2.1 showed that the LDA classifier is the most popular for different sensory systems, either in its original form or pending some improvements. This is due to the fact that it combines a low computational cost and classification accuracy being similar to the one obtained with more greedy non-linear classifiers [179]. Lowering the computational cost has a direct impact on the achievable prediction time and sampling rate. The highest prediction time has been achieved by combining a LDA classifier with a dedicated computing engine, namely an FPGA. However, SVM holds promise to outperform LDA in accuracy [42, 102], pending a more greedy computational load. We did not find papers reporting SVM development on a dedicated engine like FPGA. Capacitive sensing have been developed by a small community. Therefore, there is still some work to be done to estimate their exact prediction capacities, the most appropriate classifiers, and the potential development in a dedicated engine.

For systems using exteroceptive sensors, terrain detection with a single beam ranging sensor did not achieve high prediction time in comparison to systems which use wider range sensors such as LIDAR or depth camera. However, high prediction time was recorded with single beam ranging sensor when these sensors are mounted on moving parts of the human body in such way that they can scan the environment [41]. Terrain detection by computing the difference between depth images did not lead to high prediction time. However, depth camera achieved higher prediction time using point cloud or depth data based algorithms. The following paragraphs

review the main features of terrain identification systems across both categories and highlight their respective pros and cons.

Prediction time: Prediction time results provided in Tables 2.1 and 2.2 clearly show that systems with exteroceptive sensors outperform those using proprioceptive sensors. Prediction time of systems using proprioceptive sensors is strongly influenced by several user-related and gait-related factors such as the leading leg and transition type. For instance, previous research showed that the prediction time of the transition from level ground to stair ascending is negative when the sound leg is leading and positive when the prosthesis is leading [84]. Moreover, such systems typically displayed different prediction times for different transitions when tested with the same user, such as for the same transition with different users. These differences can reach up to 500 ms [78,84]. Systems using exteroceptive sensors, i.e. with high prediction capacity, are typically less or not affected by such factors.

Detection accuracy: The vast majority of systems reviewed in this chapter, and in particular in Tables 2.1 and 2.2, achieved high accuracy. However, this statement has to be mitigated since accuracy might also be sensitive to the user, transition type, and even gait phase. For instance, [42] reported 99% accuracy for terrain detection conducted during the stance phase and 95% during the swing phase of walking. Again, this sensitivity to user- and locomotion-related features is more prominent in systems using proprioceptive sensors, i.e. displaying low prediction capacity.

Reliability: Proprioceptive and exteroceptive sensors are respectively sensitive to variations in body and environmental conditions. Exteroceptive sensors like depth sensors are affected by sunlight while stereo cameras cannot work in low contrast or dark environment. As a result, some of them can work only in either indoor or outdoor environments. Proprioceptive sensors are robust against lighting and weather conditions, so they can be used in a larger spectrum of environmental conditions than those relying on vision. Reliability is also depending on the capacity of the different systems to work whatever the user's dress code. Indeed, systems having no vision sensor, i.e. with low predictability, can be hidden or worn under the clothes, while this is typically more challenging for systems using exteroceptive sensors. The body condition itself might have an impact on reliability. For instance, lower limb amputees do not display the same muscle activity as intact people. Consequently, EMG sensors would convey different patterns as a function of the level of amputation, skin condition and socket loading [91]. Similarly, signals recorded by contact capacitive sensors will tend to drift because of slip of the sensory bands due to sweat and movements. This will eventually cause wrong terrain estimation if the system is not periodically re-calibrated. These body-related factors have thus a stronger influence on in systems using proprioceptive sensors. Finally, the learning curve that is typically displayed by people using an assistive device such as a powered artificial leg might also have an impact on reliability. Again, a gait recognition algorithm based on proprioceptive sensors would have to be periodically re-calibrated following the user's learning curve [102]. However, it is likely that systems based on vision sensors, i.e. with high prediction capacity, would be less sensitive to changes in locomotion patterns. Hence, such periodic re-calibration training may not be necessary.

Usability: Systems with exteroceptive sensors can provide an accurate modeling of the terrains in front of the subject. Therefore, they provide a surrogate of vision to visually impaired people in order to help them achieve their path planning, i.e. plan-

ning of their locomotion trajectory. Indeed, vision-based sensing can be translated into a signal that is perceived by an intact sensory modality of the user. For instance, an audio tone can be delivered to convey several features such as terrain type or distance to obstacle. Interestingly, these vision-based strategies can be swiftly applied to assist not only visually impaired people, but also lower limb amputees and people with gait disorders. In particular, terrain modeling turns to be critical for adapting the control strategy of an assistive locomotion device at the right time, i.e. just when the terrain transition is faced. In contrast, proprioceptive sensors can be deployed only for people having both an intact exteroceptive system – and thus managing their own path planning – and capable of producing voluntary muscle contractions and/or movements; this is why they have been developed mostly to assist people with lower limb amputation. In contrast, people with lower limb amputation may not prefer using exteroceptive sensors which are mounted on the user body (above the clothes) since they may not like their assistive system to be visible.

Computational load: Systems using exteroceptive sensors need higher computational power than systems which use proprioceptive sensors only. In particular, vision based systems with depth camera provide a large amount of data about the environment, and this data has to pass through multiple processes for detecting the corresponding terrains. This has a negative impact on the size of the hardware and battery that the user has to carry when using such assistive systems. In contrast, simpler data are obtained from proprioceptive sensors, and these data can be classified with single layer classifiers (e.g., LDA and SVM) with smaller and lighter hardware. Moreover, systems with proprioceptive sensors using these classifiers achieve comparable classification performance as more complex classifiers such as artificial neural networks (ANN) which need more computational resources [42, 71].

Table 2.2: Overview of systems with high prediction capacity. Columns 5 and 6 indicate whether these methods were tested for indoor and/or outdoor environments. The last three columns report the maximum distance for detecting a terrain change, the average prediction time, and detection accuracy – when available – for the representative task of transitioning from overground walking to stair climbing. $T_{pred} > 0$ captures that all these methods detected the transition before it actually happened

References	Targeted Terrains	Sensor	Algorithm	Indoor	Outdoor	Max distance [m]	T_{pred} [ms]	Accuracy [%]
[109]	Stairs	Ultrasonic sensor	SVM classifier	✓	✓	1	910*	72.41
[110, 111, 113]	Stairs and slopes	Laser ranging sensor	classification tree	✓	✓	\	692.68±74	98
[41]	Stairs	FMCW radar	Filtering and clustering	✓	✓	5	4500*	\
[117]	Stairs	LIDAR	Classifying the points into horizontal and vertical segments, and filtering based on segment length	✓	✓	4	3600*	95
[140, 141]	Stairs	RGB-D	Stairs lines detection in depth image with Hough transform, and SVM for up and down-stairs recognition (2D-based algorithm)	✓		3.5**	3153*	97.2
[135, 136]	Stairs	TOF camera	Feature extraction of depth difference image with SVM classifier (2D-based algorithm)	✓	✓	1	910*	94.1
[132, 154, 155]	Stairs	RGB-D	RANSAC for plane fitting, stairs detection based on difference in height between planes (3D-based algorithm)	✓		3.5**	3153*	97.33
[153]	Stairs	RGB-D	RANSAC for Plane detection, and concave hull algorithm for steps recognition (3D-based algorithm)	✓		3.5**	3153*	95.8
[43, 167]	Stairs and slopes	TOF camera	Dimensional reduction, and neural network for classification (3D-based algorithm)	✓	✓	\	3340	98.5
[169, 170]	Stairs and slopes	RGB-D	Evaluation/histogram map construction (3D-based algorithm)	✓		3	2700*	\
[156, 157]	Stairs	RGB-D	Gradient computation Directional thresholding (3D-based algorithm)	✓		1.5	1351*	98.8
[151]	Stairs and slopes	RGB-D	Points classification according to their normal (3D-based algorithm)	✓		3.5**	3153*	\
[149], [160], [161]	Uneven terrains	RGB-D	Unevenness Points Descriptor (3D-based algorithm)	✓		3.5	3153*	\
[143]	Stairs	Stereo camera RGB and depth cameras	Edge detection, one dimensional feature extraction from depth image, and classification with SVM (Fusion of RGB and Depth)	✓		22	19820*	\
				✓		2	1800*	92.7

*Estimated from the maximum detection distance if the wearer would walk at 4 km/h.

**Estimated based on the effective depth range which is up to 3.5 meter as determined in [155].

2.5 Conclusion

In this chapter, we reviewed many sensory systems that were developed to identify terrains during locomotion tasks, and in particular in order to identify the locomotion intention of a disabled user across different possible tasks (flat ground walking, stairs ascending/descending, etc.). We sorted these systems into two categories, i.e. these deploying proprioceptive sensors only, and those further relying on exteroceptive sensors like vision-based devices. This last category is only emerging in the field of locomotion assistance, but we further reviewed contributions from mobile robotics that could be adapted to assist human users. This review brought the conclusion that exteroceptive sensors are necessary if the assistive device requires to anticipate the locomotion transition by more than one step, i.e. what we denoted to correspond to high prediction capacity. Owing to the emergence of active assistive devices for locomotion like exoskeleton or bionic prostheses, it is likely that such proprioceptive sensors will emerge as a key ingredient to promote synergetic behavior between the user and the device in different locomotion contexts. The chapter ended with a discussion highlighting specific challenges related to these exteroceptive systems, such as usability and reliability.

Early terrain detection system implementation

3

Related publication:

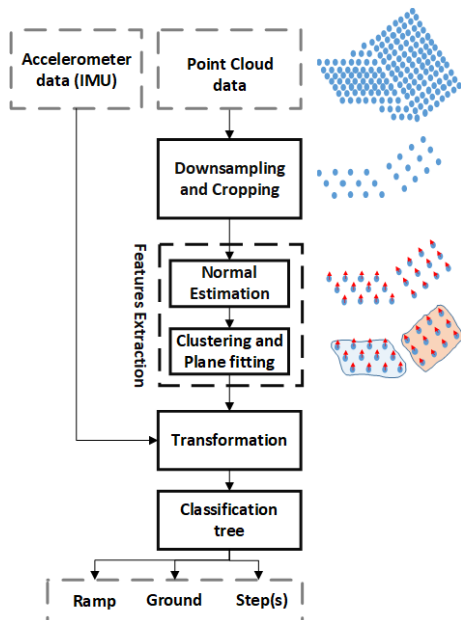
- **Al-dabbagh, Ali HA**, and Renaud Ronsse, *A New Terrain Recognition Approach for Predictive Control of Assistive Devices Using Depth Vision.*, International Symposium on Wearable Robotics. Springer, Cham, 2020.

3.1 Introduction

Based on the information introduced in the review, we could observe that systems with exteroceptive sensors overcome the systems with proprioceptive sensors in many important aspects. For instance, they have a higher prediction capacity which gives a sufficient amount of time for the prosthesis or exoskeleton to react. Moreover, these systems rely on environmental perception through analyzing the captured information with a camera or ranging sensor, and this makes such systems to be users independent. Therefore, they can be used to assist a wider range of disabled people including the visually impaired.

Systems with exteroceptive sensors usually use a wearable camera or ranging sensor to capture information about the surrounding environment. However, many types of these sensors have some limitations to be used in our application. For instance, RGB cameras suffer from some drawbacks regarding their usability in everyday life contexts. Firstly, they do not work in dark environments. Secondly, more advanced features like step height or ground inclination can hardly be estimated from RGB images which prevents the ability to finely modulate the desired joint torque patterns as a function of environmental features. Also, RGB cameras cannot accurately locate the detected terrains with respect to the user which may cause a too late or too early reaction of the assistive device. Therefore, relying on sensors that could build a 3D image of the desired terrain is a more solid approach to recognize the affordances and extract environmental features. Constructing such an image is achievable by using a 3D-LIDAR or depth camera. The former is usually bulky, expensive, and power demanding since it may have movable mechanical parts inside,

Figure 3.1: Block diagram illustrating the overall terrain detection algorithm. Point cloud data acquired from a depth camera and its orientation obtained from an IMU sensor are its inputs. It provides a detection of the locomotion terrain(s), i.e. flat ground, stairs, and/or ramp, as output.



while the latter has a more compact size, cheaper, and consumes less power which makes it more suitable for wearable applications. In our system, we implemented an algorithm that relies on point cloud processing for extracting environmental features from the captured 3D image toward reaching terrain detection.

In this chapter, we present the first implementation of our terrain detection algorithm. It starts with exploring the steps applied to the point cloud features extraction and terrain identification. Then, we go through some conducted experiments with presenting and discussing the achieved results.

3.2 Terrain detection algorithm

We implemented a computer vision-based algorithm for terrain recognition using a depth camera. The algorithm processes the point cloud data to obtain the detected terrains for the whole detection range. The raw point cloud data obtained from the depth camera passes through multiple processing steps to extract environmental features, and thus achieve terrain detection as shown in Fig. 3.1. We will go through each step in the following sections.

3.2.1 Downsampling and Cropping

This step aims at reducing the computational load by minimizing the number of points in the cloud. Indeed, the high density of point cloud data causes processing delay, and not all the points bring a significant part of new information. The point cloud resolution is downsized by using a 3D voxel grid filter, with a leaf size of 3 cm [152]. Although there are other filtering techniques found in the literature —such as Normal-based Bilateral Filter (NBF), Moving Least Square (MLS), and Weighted

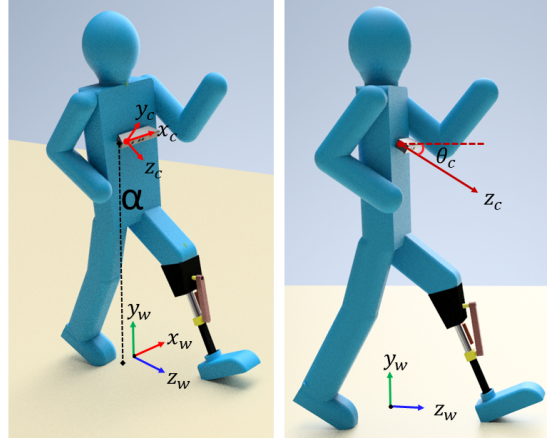
Locally Optimal Projection (WLOP)— it is proved that 3D voxel grid filter is more efficient than these other ones in terms of computational cost, with nearly two or three orders of magnitude faster [180]. Moreover, it provides an accurate representation of the underlying surface within the filtered cloud [180]. To further reduce the computational cost, a region of interest (ROI) of one meter wide in front of the user was specified in order to eliminate data lying out of this region, similarly as in [155].

3.2.2 Features extraction

In this step, we extract environmental features of each group of the point cloud from the previous step. This is achieved through the following sequence of processing:

- For each point, a normal vector is estimated from its neighboring points [157] based on Principal Component Analysis (PCA) [119, 137, 151]. This approach is efficient and has been used in other systems [149, 160, 161]. It computes the eigenvalue and eigenvector of a covariance matrix created from direct neighboring points. The normal direction is estimated as the eigenvector having the smallest eigenvalue [137]. The point cloud data is then clustered or segmented into multiple homogeneous group. The points in each group will have the same properties. There are many techniques found in the literature for that purpose. For instance, machine learning based cloud segmentation approaches have been used in many robotic applications, and they demonstrate the ability to accurately segmenting the point cloud even in the complex scenes [181]. However, these methods require a large computational cost and might not be suitable for our needs [182]. Alternatively, Edge-based segmentation method was presented by [183]. This method uses edge detection to outline the borders of various region and group the points inside each region to define the final segment. While this technique allows fast cloud clustering, it can fail in the presence of noise and uneven density of the point cloud [184]. In contrast, Region based methods are more robust to noise than Edge-based segmentation [182]. They rely on neighborhood information to gather nearby points that have similar properties in one cluster. There are two versions of those techniques seeded and unseeded [182]. A typical drawback of the unseeded version is that it needs prior knowledge about the number of regions and object models which can be unknown in complex scenes [182]. Therefore, the seeded version of region growing method is applied in our pipeline. It starts from a point with minimum curvature called *seed*, and expands the region to include neighboring points having quasi-parallel normals and similar curvature [119]. When the region cannot be expanded anymore, a new seed is chosen, and the process is repeated until a pre-specified threshold is reached. This algorithm outputs a set of points (cluster) belonging to the same smooth surface. The region growing algorithm has a good physical relevance since the planes are detected from a closed region depending on single element (seed) rather than from a set of uncorrelated points distributed in the scene [119, 137].
- Finally, a plane is fitted over each cluster using the RANdom SAmple Consensus (RANSAC) algorithm developed by [185]. Then, for each fitted plane, its normal vector $\mathbf{n}^c = (n_x^c, n_y^c, n_z^c)^T$ and centroid $\mathbf{o}^c = (o_x^c, o_y^c, o_z^c)^T$ are com-

Figure 3.2: The left panel shows the relationship between the camera frame (x_c, y_c, z_c) and the local world frame (x_w, y_w, z_w) . The right panel shows the orientation of the camera frame with respect to the horizontal plane, in the sagittal plane.



puted defining the plane orientation and location in the 3D environment, with respect to the coordinate frame attached to the camera (x_c, y_c, z_c) (see Fig. 3.2)

3.2.3 Features transformation

Point cloud data acquired from a depth sensor are provided in a coordinate frame being centered on the camera, and this causes issues for proper terrain detection. For instance, if the camera points downwards, the level ground will be detected as a ramp [43]. A common solution is to transform the reference coordinate system to the ground or local world coordinates based on the measured instantaneous orientation of the camera. This orientation can be provided e.g. by an IMU that is attached to the camera. Typically, this transformation is applied to the whole point cloud just after acquisition, and thus before feature extraction [43, 137, 156]. Here, we applied the transformation to the obtained plane features (centroid and slope), since it is not necessary to rotate and translate all points of the acquired cloud. Fig. 3.2 defines the coordinate frame attached to the camera (x_c, y_c, z_c) and the so-called "local world frame" (x_w, y_w, z_w) that is used for terrain recognition. This local world frame is moving with the user but is kept with y_w pointing upwards and z_w pointing in the user's direction of locomotion. Assuming that the camera is only translating and rotating in the $x_w - z_w$ plane, the relationship between both frames is thus given by the homogeneous transformation matrix:

$$\mathbf{H}_c^w = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta_c & -\sin \theta_c & \alpha \\ 0 & \sin \theta_c & \cos \theta_c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

where θ_c is the camera rotation angle and α is the distance between the ground and the camera, as represented in Fig. 3.2. Accordingly, the plane orientation and centroid can be transformed back to the local world frame by pre-multiplying their coordinates with the transformation matrix, i.e. $(n^w, 1)^T = \mathbf{H}_c^w (n^c, 1)^T$ and $(o^w, 1)^T = \mathbf{H}_c^w (o^c, 1)^T$.

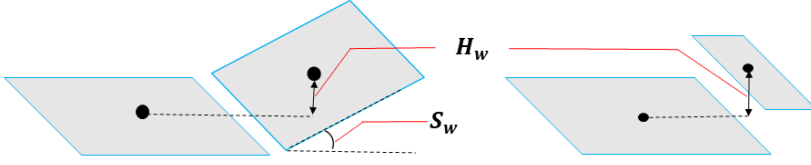


Figure 3.3: Features fed to the classification tree to achieve various terrain recognition. S_w is the plane orientation angle, H_w is the plane height with respect to the user's feet.

3.2.4 Classification tree

The following features are extracted for each plane whose coordinates have been transformed in the local world frame during the previous step:

- $S_w = \text{atan}(n_y^w / n_z^w)$ is the plane orientation angle, defined as the angle between the plane normal vector and the vertical axis y_w .
- $H_w = o_y^w$ is the coordinate in the plane centroid in the vertical direction, i.e. the plane height with respect to the user's feet.

The features passed to the classification tree are pictured in Fig. 3.3. The tree applies several tests to these features which leads to various terrain predictions as shown in Fig. 3.4. It starts by examining the features against ramp, stair, and finally flat ground candidates. For instance, the detected plane is classified as being a ramp up if $5^\circ < S_w < 25^\circ$ and $0.1\text{m} < H_w < 1\text{m}$. If this up-ramp candidacy test is not positive, then the plane is examined against down-ramp conditions, etc. Planes having feature values lying outside the conditions of ramp, stairs, and flat ground terrains are considered obstacles and labeled as "No Detection" (ND) by the algorithm. A specific color is assigned to each detected type of terrain for visualization purposes.

3.3 Experimental protocol

3.3.1 Software

The algorithm was implemented in the Robot Operating System (ROS) and using pre-existing Point Cloud Library (PCL) [152]. The system initially was tested in a simulated environment using (Gazebo) software with different geometrical terrains and sensor configurations.

3.3.2 Hardware

A wearable tool was designed for the purpose of data collection (see Fig. 3.5). The tool consists of an IMU (9DoF Razor-SparkFun), a Kinect camera (Xbox 360), and minicomputer (Jetson Nano). This minicomputer performs the first step of the algorithm (downsampling and cropping) and records downsampled points cloud, raw IMU data, and RGB data being used for off-line labeling of the ground truth of the detected terrain. The data are recorded with different sampling rates (5fps for the

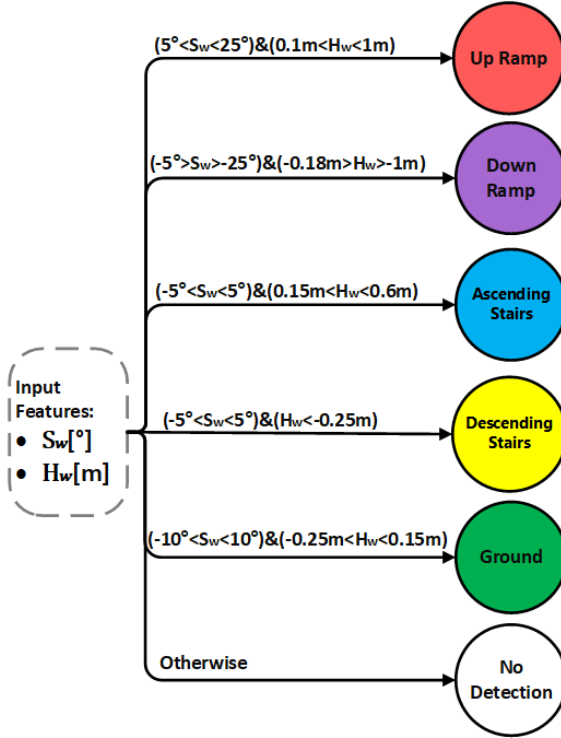


Figure 3.4: The classification tree examines the extracted features S_w , H_w against several conditions to achieve terrain identification. First, it begins by testing the features for the ramp conditions, then for the stair conditions, and finally for the ground conditions. The plane is considered to be an obstacle if its features lie outside all the above constraints. Colors given for each type of detected terrain correspond to the colors assigned to the raw point cloud data for the visualization purpose. (See Table 3.1)

point cloud data, 200 fps for IMU data and 30fps for the RGB data). The wearable tool is designed to be mounted on the subject chest with belts. This sensor configuration tends to provide more stable images than other configurations such as the one on leg or head [119, 137].

3.3.3 Experiments

The system was tested with six healthy subjects (4 males and 2 females aged between 27-32 years) by mounting the sensor and processing unit on the subject's chest with the wearable tool. The subjects were asked to follow a pre-specified indoor path that contains a stair and ramp on the UCLouvain campus as illustrated in Fig. 3.6. During the test, the path was cleared from obstacles and people. In order to estimate the system accuracy in detecting each locomotion mode in an uncluttered environment, the recorded data with the 6 subjects were categorized based on the five locomotion tasks identified above and labeled as ground (G), down-ramp (DR), up-ramp (UR), as-

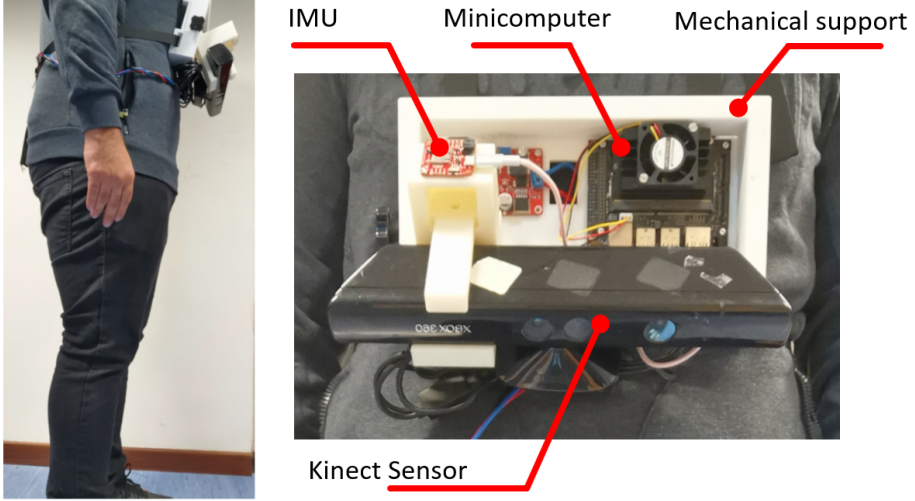


Figure 3.5: The designed wearable tool to carry the system hardware consisting of a depth camera, a mini computer, and an IMU.

Table 3.1: Detection labels given to each type of locomotion affordances, together with the color codes used for visualization purposes.

Locomotion affordance	Label	Color code
flat ground	G	● Green
down ramp	DR	● Purple
up ramp	UR	● Red
ascending stairs	AS	● Blue
descending stairs	DS	● Yellow
others (no detection)	ND	—

ending stairs (AS) and descending stairs (DS). Ground truth labeling was performed using the RGB footage recorded during the experiments. Finally, we conducted some experiments to compare the computational cost between the features transformation approach and all cloud transformations at different cloud densities.

3.3.4 Accuracy

The detection accuracy was computed for each locomotion mode as:

$$A_c = \frac{L_{positive}}{L_{total}}, \quad (3.2)$$

where A_c refers to the accuracy, $L_{positive}$ is the number of true positive samples, and L_{total} is the number of total collected samples that is equal to:

$$L_{total} = M_{fps} \times t_n, \quad (3.3)$$

where M_{fps} is the frame publishing rate by the algorithm, and t_n is time taken to complete a portion of the full path that is specific to a given locomotion mode.

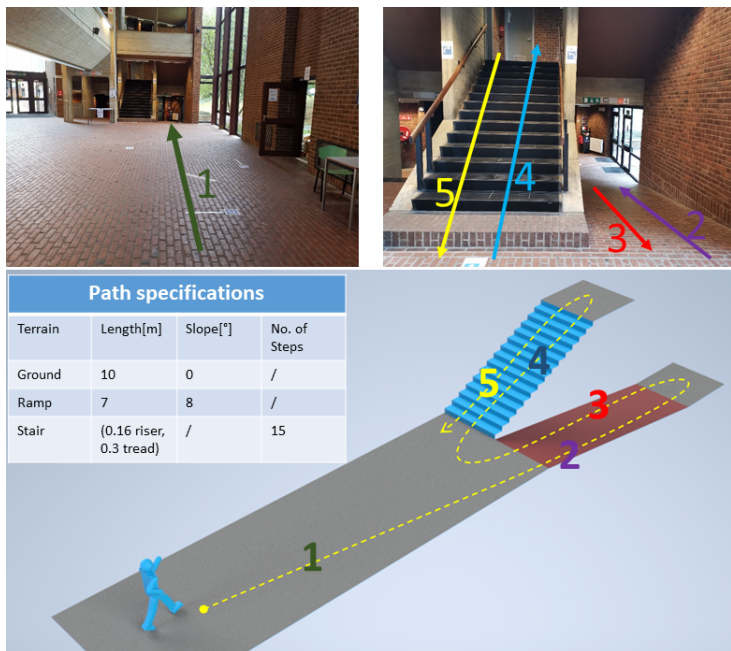


Figure 3.6: Top pictures are snapshots of the real experimental environment. Each subject walked on the path with the wearable tool starting from 1 (flat ground) and finishing with 5 (downstairs). The bottom panel shows a 3D representation of the path with its specifications

3.4 Results and discussion

Samples of the recognition results in the simulated environment are displayed in Fig. 3.7. Using Kinect, some samples captured in a real environment are shown in Fig. 3.8. The average detection accuracy for different locomotion modes is shown in Fig. 3.9, and Table 3.2 displays the confusion matrix. Zeros in such a matrix implies that the corresponding locomotion mode was never predicted when the user was walking over the specific actual terrain. Our method reached above 90% of average detection accuracy for all locomotion modes while the grand average accuracy was above 94%. The lowest accuracy was recorded during the DR mode. This locomotion mode was mostly confused with the leveled ground as evidenced by the confusion matrix. In the second set of experiments, we quantified the computational cost of our algorithm, with a fixed camera on the same scene. We further compared it with a more standard approach consisting of rotating the point cloud back to the inertial frame before the feature extraction. Based on multiple trials, we observed that our method is about 20% cheaper in computational resources than an "all cloud transformation" method, at least for cloud sizes above 10^5 points.

However, the algorithm still has some limitations regarding distinguishing between ground and ramp affordances at the transition region. Such separation is important for achieving accurate terrain identification for several steps. Also, the detection results still need further processing to predict the upcoming locomotion

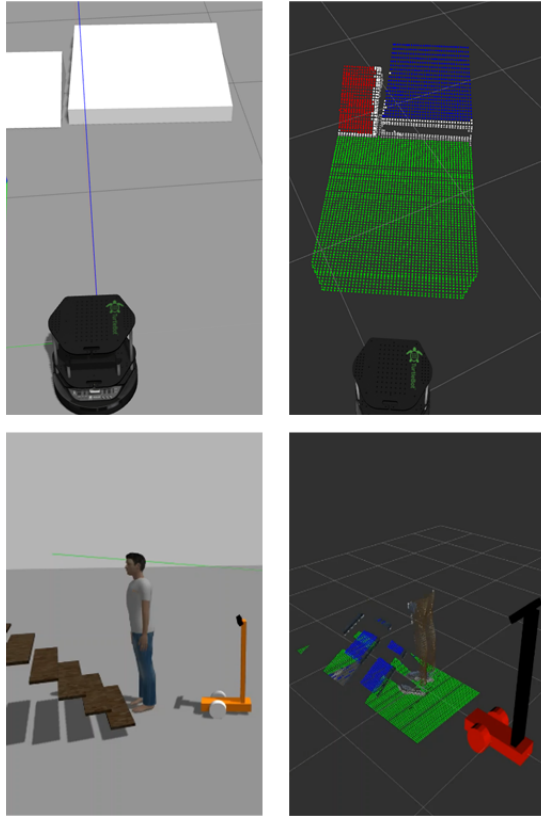


Figure 3.7: Images in the left column show wheeled robots carrying depth camera in virtual environment with different sensor configurations. The right column images show detection results after processing the point cloud obtained from the depth sensor. Colors are assigned to each type of the detected terrain for visualization purposes. The point clouds for each terrain type are color labeled with the labels given in Table 3.1.

modes for some steps.

Table 3.2: Confusion matrix of the recognition results. Values are given as percentages.

	G	DR	UR	AS	DS	ND
G	98.14	0	1.85	0	0	0
DR	9.54	90.45	0	0	0	0
UR	0	0	99.5	0.5	0	0
AS	2.62	0	0.32	95.08	0	1.96
DS	6	0.8	0	0	93.14	0



Figure 3.8: Detection samples of five locomotion captured with Kinect sensor in real environment. The point cloud for each terrain type are colored labeled with the labels given in Table 3.1.

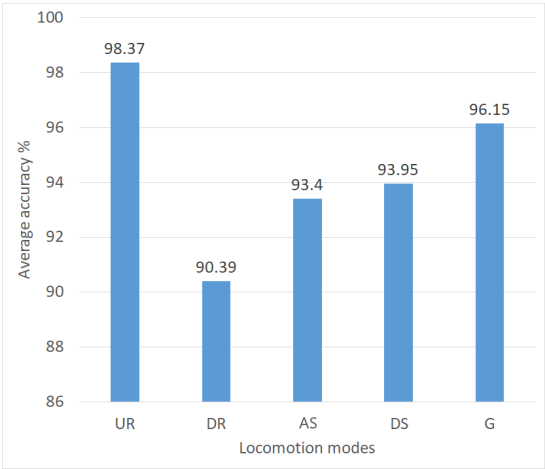


Figure 3.9: Average accuracy for different locomotion modes for the all six subjects (UR: up ramp, DR: down ramp, AS: ascending stairs, DS: descending stairs, and G: ground-level).

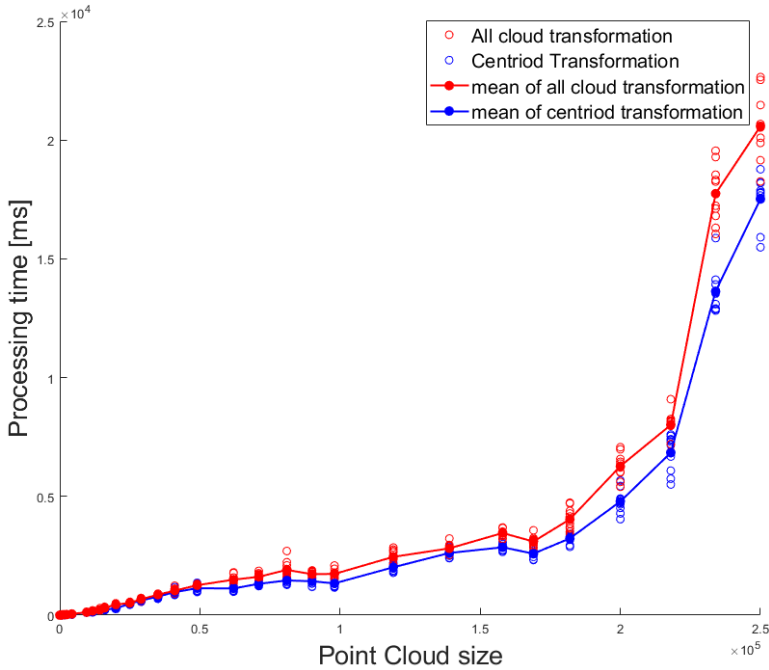


Figure 3.10: Processing time for various point cloud sizes. Two different transformation methods were tested: red points denote trials where all points of the cloud are rotated in the world frame before feature extraction, while the blue points correspond to our approach, where we extract features in the camera frame and rotated only these extracted features (plane centroid and orientation). Each dot stands for one recorded run time per frame. The plain lines connect the mean of recorded run times

3.5 Conclusion

In this chapter, a new terrain detection algorithm based on a depth-sensing camera is presented. The system detects the main ground, steps, and ramps with above 90% accuracy. It requires less computational time because we only transformed the plane features rather than the whole point cloud. Consequently, this approach is promising for wearable systems with limited processing capabilities. However, the algorithm still needs further improvements in order to be able to predict the upcoming locomotion modes for multiple steps in advance. Due to the limited Kinect sensor capability, the system was only tested in indoor environments. However, further testing is required in several indoor and outdoor environments in order to precisely quantify its detection abilities. Therefore, the next chapter will focus on addressing these issues and testing the algorithm detection ability in partially occluded paths. This is important since real urban environments will likely be cluttered with other people or obstacles.

Terrain detection system for multiple steps with a fixed step size

4

Related publications:

- **Al-dabbagh, Ali HA**, and Renaud Ronsse. "Using Depth Vision for Terrain Detection during Active Locomotion." *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021.
- **Al-dabbagh, Ali HA**, and Renaud Ronsse. "Depth Vision based Terrain Detection Algorithm during Human Locomotion." *IEEE transactions on medical robotics and bionics* 2022 (in press).

4.1 Introduction

A typical depth camera can record environmental information for more than one meter of distance. Therefore, it is possible to exploit such information to estimate the locomotion terrain several steps in advance, and this would be relevant to handle smooth and timely transitions with an active assistive device. In this chapter, we present an improved version of the depth vision-based terrain recognition algorithm introduced in the previous chapter. The new version can predict the user locomotion mode up to three steps ahead of time. More precisely, our algorithm estimates the probability of walking on different types of terrains (ground, stairs, slope) for the upcoming three steps. The geometrical features of the detected terrains are also computed and linked to the upcoming three steps.

The present work is an extended version of a previously published conference paper [186] which introduces a classification tree-based terrain detection algorithm for several steps (see Appendix A for more information about the experimental methods and results achieved with that version). With the extended version presented in this chapter, the hardware has been updated with a more compact depth camera that

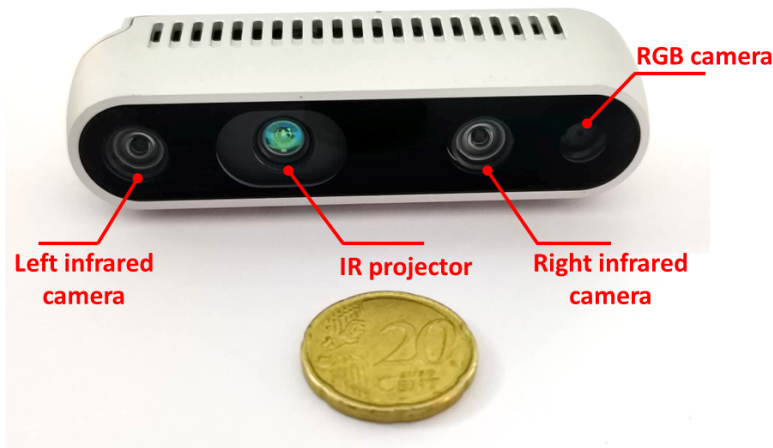


Figure 4.1: Realsense D435i stereo vision based depth sensor. It has two infrared cameras, a single RGB camera, and an IR dotted pattern projector which improves the overall camera performance by adding more light and texture.

can work under ambient sunlight and the classifier has been changed to a more efficient one. Moreover, the system is now validated with several paths in indoor and outdoor environments even with the partially occluded path which is ubiquitous in urban contexts.

This chapter is organized as follows. Section 4.2 provides a description of the implemented algorithm. Section 4.3 details the experimental protocols, hardware used for validation, and the accuracy estimation method. Sections 4.4 and 4.5 report and discuss the experimental outcomes and provide a general overview regarding the system performance in terms of accuracy. Finally, the chapter ends with a conclusion.

4.2 Terrain detection algorithm

The method outlined in this chapter achieves terrain identification from data obtained by a camera providing depth information from the environment composing the field of view. This information is represented as a cloud of data points capturing the environment structure, each point located relative to the camera lens. Several depth camera technologies exist in the market, on top of which are those working either with structured light, time of flight, or stereo depth cameras. Although stereo vision cameras are slightly less accurate at determining an object's position in a scene [187], they are robust to the ambient sunlight in the environment [188]. They can thus work in indoor and outdoor environments. Consequently, we selected a Realsense D435i (Intel, US) stereo-based depth camera for the terrain recognition system. This camera has a compact size (90 mm × 25 mm × 25 mm) and requires low power consumption (3.6W). It embeds a 6-DOF IMU, with 0.04° standard deviation noise for the pitch angle at the static condition. It is thus particularly suited for wearable applications (see Fig. 4.1).

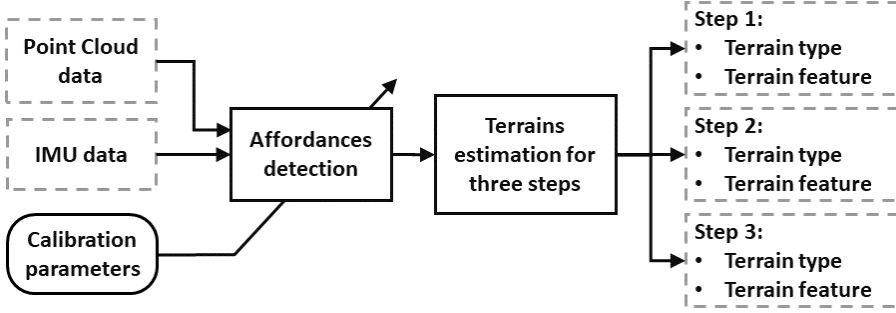


Figure 4.2: Overall terrain detection algorithm. It takes point cloud and IMU data as inputs, and provide terrain type prediction and features estimation for three upcoming steps. Our approach has two main stages: the first stage recognize affordances type while the second stage estimates the locomotion modes for the three upcoming steps. The algorithm also computes geometrical terrain features for each step.

The raw point cloud data obtained from the camera pass through multiple processing steps to extract environmental features in order to achieve terrain detection. Our algorithm has two main stages (see Fig. 4.2) that are executed at each time step. In the first stage, so-called locomotion affordances are identified, together with their features. In the second stage, the terrain type for the three upcoming steps is estimated and displayed alongside their features. In the following subsections, we will go through each of these stages in detail.

4.2.1 Affordances detection

This stage consists of passing the raw point and IMU data through several steps in order to extract features and detect affordances, as illustrated in Fig. 4.3. Some of these steps are already explained in detail in the previous chapter; therefore, we will go throw them in a brief way focusing on the differences that exist in this version of the algorithm.

4.2.1.a Downsampling and Cropping

The Realsense depth camera publishes the raw point cloud with a density of about 1mm of the distance between neighboring points. In this algorithm, the point cloud resolution is downsized using a 3D voxel grid filter, with a leaf size of 4cm [152]. As previously done, a region of interest (ROI) of one meter wide in front of the user is specified in order to eliminate data lying out of this region.

4.2.1.b Cloud split and merge

This step is useful to identify the transition edges between grounds of different inclinations. Indeed, it enables the algorithm to separate the physically connected planes that have different orientations. This can lead to enhance the robustness of the method by accurately predicting the upcoming locomotion mode at transition

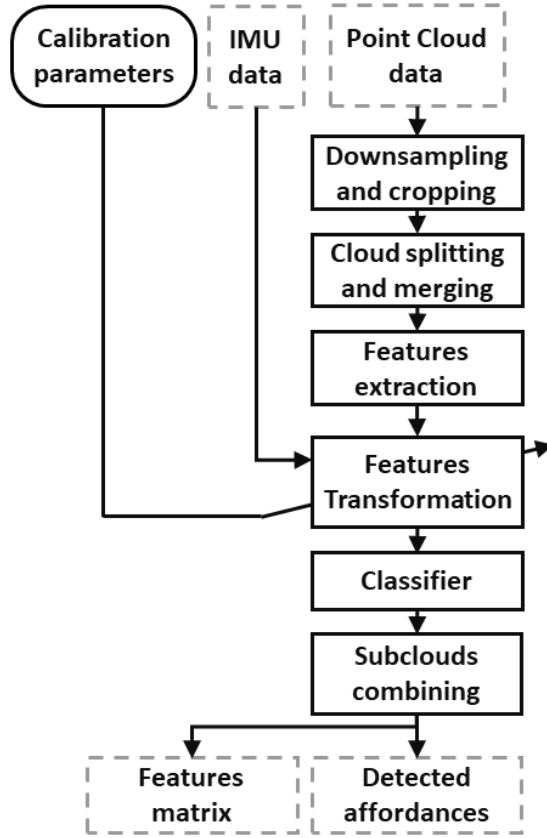


Figure 4.3: Affordances detection stage. The raw point cloud passes through several processing steps in order to achieve detection of locomotion affordances, and their geometrical features.

regions that are physically connected, such as up/down ramps and flat ground. First, the whole point cloud is divided into three subclouds as a function of their closeness to the camera. About 700 points are thus kept for each one. Then, a plane is fitted over each subcloud using the RANdom SAmple Consensus (RANSAC) algorithm [185]. The global normal vector of this plane is then estimated. If the orientation difference between the normal vector of two adjacent subclouds is found to be less than 5 degrees, then both subclouds are assumed to belong to the same physical plane and are thus merged. Otherwise, they are kept separated for the following steps. Consequently, this step outputs between one and three subcloud(s) which are all passed to the next steps to be processed separately (see Fig. 4.4).

4.2.1.c Features extraction

This step aims at extracting the features of different affordances in the environment. First, for each point in a subcloud, its normal vector is estimated from the neighboring points [157]. Then, the subcloud points with their normals are passed into the region

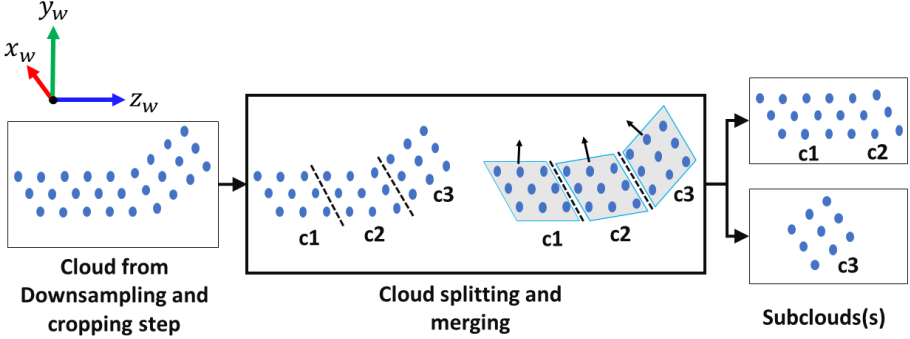


Figure 4.4: The downsampled cloud is divided into three subclouds (c1, c2 and c3) based on their closeness to the camera. In this example, c1 and c2 are merged together since their global normal difference is less than a pre-specified threshold, while c3 is kept separated since its orientation difference with c2 is larger than the threshold.

growing algorithm to group the points included in each subcloud into several clusters based on the orientation of their plane. This algorithm outputs clusters of points, each of them belonging to the same smooth surface. Points that are not belonging to a cluster are withdrawn from the subcloud. This process is essential to separate different affordances contained in each subcloud.

Finally, a plane is fitted over each cluster contained in a given subcloud using again the RANSAC algorithm [185]. Then, for each fitted plane, its normal vector $\mathbf{n}^c = (n_x^c, n_y^c, n_z^c)^T$ and centroid $\mathbf{o}^c = (o_x^c, o_y^c, o_z^c)^T$ are computed and thus define the global plane orientation and location in the 3D environment, with respect to the coordinate frame attached to the camera (x_c, y_c, z_c) , see Fig. 3.2.

4.2.1.d Features transformation

The extracted features are transformed from the camera frame (x_c, y_c, z_c) to the local world frame (x_w, y_w, z_w) (see Fig. 3.2). The relationship between both frames was given by the homogeneous transformation matrix in equation 3.1. The distance α and the zero of the angle θ_c are assumed to be constant and are thus calibrated once before the feature transformation step, see Fig. 4.3. In particular, α is measured on the participant when standing still for a few seconds before each experimental trial, and θ_c is zeroed by computing the mean camera orientation measured with an IMU, when the participant is walking on flat ground during 4 seconds. Then, selected features are computed for each transformed plane (see Fig. 4.5):

- $S_w = \text{atan}(n_y^w / n_z^w)$ is the plane orientation angle, defined as the angle between the plane normal vector and the vertical axis y_w .
- $H_w = o_y^w$ is the coordinate of the plane centroid in the vertical direction, i.e. the plane height with respect to the user's feet.
- D_z is the horizontal distance in the direction of locomotion z_w between the plane centroid and the camera.

4.2.1.e Classifier

In this stage, the points of each subcloud are labeled through a multistage classification process. The point cloud belonging to a plane is passed with its selected features to a two-layer classification process. First, planes with $|S_w| > 40$ are considered to be too inclined for being usable as locomotion affordances, and thus their point clouds are also labeled as "No Detection" (ND). Next, the remaining planes are fed to a trained machine learning-based classifier that classifies them into five locomotion modes as a function of their corresponding affordance that are ground (G), down ramp (DR), up ramp (UR), ascending stairs (AS), or descending stairs (DS). Their corresponding point clouds are labeled with the labels given in Table 3.1. The classified plane features corresponding to each subcloud information are eventually stored in a single matrix with each row corresponding to a labeled plane:

$$\mathbf{F}_m = \begin{bmatrix} S_{w1} & H_{w1} & D_{z1} & C_{s1} & T_1 \\ S_{w2} & H_{w2} & D_{z2} & C_{s2} & T_2 \\ S_{w3} & H_{w2} & D_{z3} & C_{s3} & T_3 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ S_{wn} & H_{wn} & D_{zn} & C_{sn} & T_n \end{bmatrix} \quad (4.1)$$

where S_{wi} , H_{wi} and D_{zi} are the plane features fed to the classifier; C_{si} and T_i are the cloud size of the corresponding plane, and the terrain label provided by the classifier, and n is the number of detected affordances in this particular subcloud.

4.2.1.f Subclouds combining

This step potentially combines the labeled subclouds only if there were more than one subcloud extracted in the splitting and merging step. Also, the feature matrices of the subclouds are stacked together in a single matrix F_m . This feature matrix rows are arranged from the closet to the farthest plane. In sum, this stage outputs a single labeled cloud of points referring to the detected affordances, and one matrix F_m denoting the geometrical features associated with the corresponding planes/affordances.

4.2.2 Terrains estimation for three steps

Based on the terrain affordances identified in front of the participant, the locomotion modes and features for the three upcoming steps are estimated as explained in the following sections.

4.2.2.a Terrain estimation for overground and ramp

This mode is used if there is no plane classified as stairs by the SVM classifier in the first stage of the algorithm. In this case, the environment in front of the user is segregated into three regions approximately corresponding to the three upcoming walking steps, with a step size equal to about 50cm. For each of these three steps, the probability of walking on a given type of terrain (G, DR, and UR) is computed as the fraction of points from the labeled cloud that is labeled with this corresponding label.

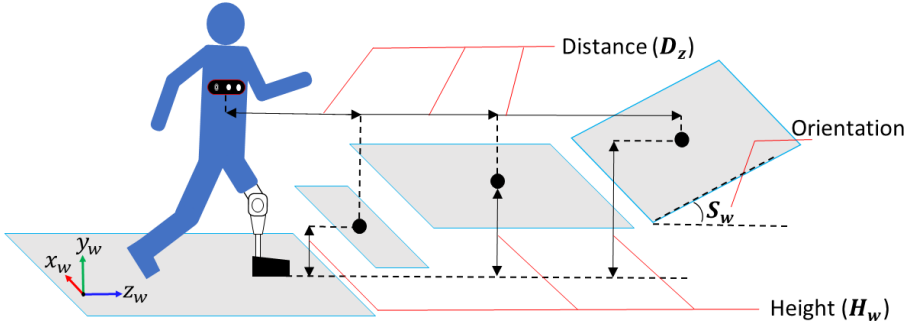


Figure 4.5: Extracted plane features: S_w is the plane orientation angle, H_w the plane height with respect to the user's feet, and D_z is the horizontal distance between the plane centroid and the camera.

As output, the algorithm thus provides the probability of walking on flat ground and up-/down-ramp for three steps upfront of the participant. The most probable terrain is simply taken as the one having the highest probability.

4.2.2.b Terrains estimation for stair locomotion mode

if there is at least one plane in front of the participant classified as ascending or descending stairs, then the following processing is applied. Using the features matrix F_m , one step is associated with one row of features. More precisely, the closest plane is linked to the first step up to the third one which, if detected, is linked to the third step. Accordingly, for each step, the system will publish the terrain type taken from the three first rows of the features matrix F_m .

4.2.2.c Terrain geometrical features

Besides estimating the locomotion mode for three steps ahead of the participant, our method can provide more detailed properties of the detected terrain. As consequence, an active locomotion assistance device could modulate its action as a function of the ground inclination or as a function of a step height. For ground and ramp locomotion modes, the angular orientation of the corresponding plane is directly provided by the feature S_w . If there are two or more planes classified as ascending or descending stairs, step elevation is calculated as the difference between two consecutive planes.

Last but not least, a median filter with a window size of three samples is applied to each terrain type information. Despite the small delay caused by this filter, it improves the robustness of the results [43]. The structure of the algorithm was implemented using Robot Operating System (ROS) [189] while the points cloud was processed with Point Cloud Library (PCL) [190]. Finally, the SVM classifier implementation and training was done using Scikit-learn [191].

Table 4.1: Paths geometrical specifications.

	Ground	Ramp		Stair	
Paths	Length[m]	Length[m]	Slope[°]	Tread[m]	Riser[m]
Indoor 1	12	7	8°	0.3	0.16
Indoor 2	12	4.5	8°	0.3	0.16
Outdoor 1	24	13.5	7°	0.3	0.16
Outdoor 2	11	21.5	5°	0.33	0.11

4.3 Experimental validation

4.3.1 Methods

Four locomotion paths were designed in order to examine the system performance, i.e. two indoor and two outdoor, see Fig. 4.6. Each path combined the five locomotion modes, arranged in different orders and with different geometrical specifications such as length, slope inclination, and step height as illustrated in Table 4.1. Eight naive and healthy participants (5 males and 3 females) aged between 24 and 41 were recruited to participate in the experiments.

3D-printed mechanical equipment was designed to hold the camera and attach it to the user's chest as pictured in Fig. 4.7. The camera was placed in order to point downwards with an angle between 45° and 55° with respect to the horizontal plane and was connected to a laptop (Alienware, core i7, NVIDIA GTX780 graphics card) by a long USB cable for data recording during experimental trials. This computer was carried by an experimenter following the participant to record the data.

4.3.2 Protocol

4.3.2.a Clear path trials

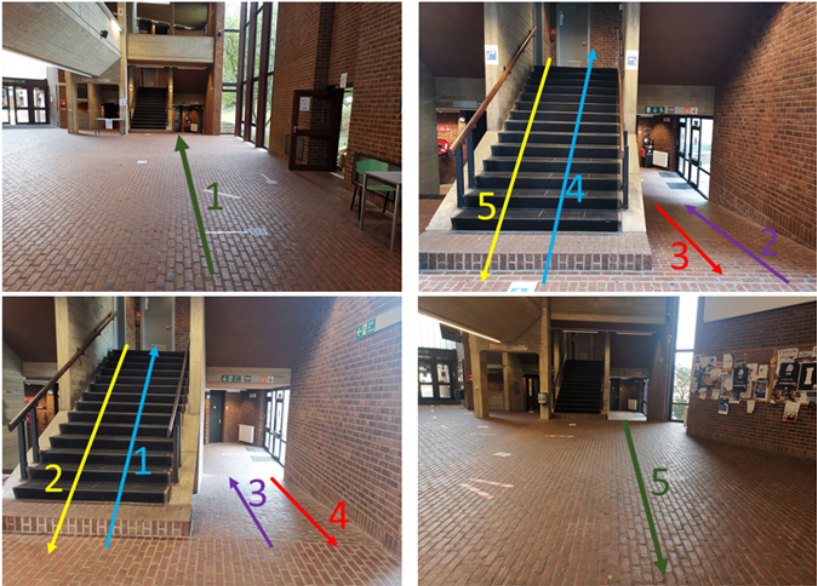
In these trials, each participant went through all four paths at their preferred walking speed while the path was free of any people.

4.3.2.b Occluded path trials

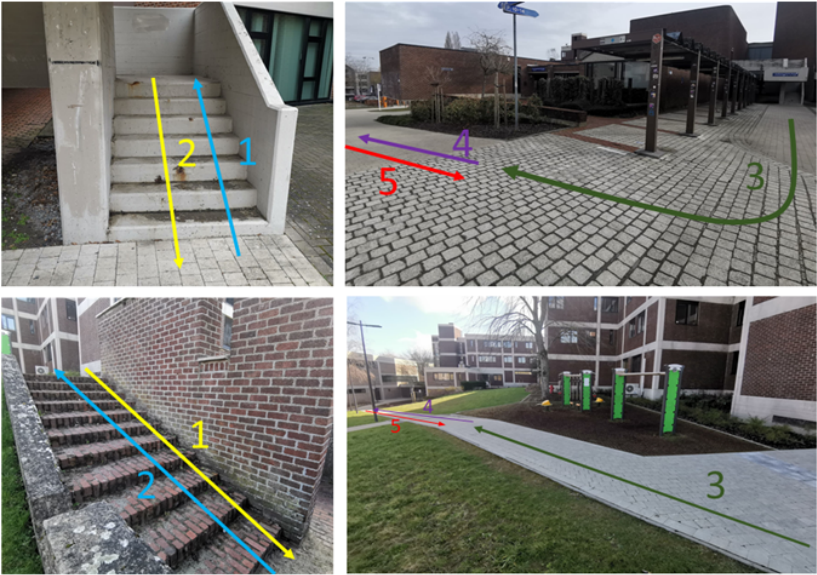
In these trials, the scene in front of the camera was partially covered by another walker walking ahead of the participant over the whole path. In order to loosely keep the distance between the participant and the leading walker occluding the path within the same range for all trials, we asked both of them to hold the extremities of a rope of length 0.9m.

4.3.2.c Recorded Data

For each trial, the computer recorded IMU data, down-sampled point cloud, and RGB image of the scene; at a rate of 15 fps. In total, we thus recorded 64 trials (8 participants $\times$ 4 paths $\times$ 2 occlusion conditions). These recorded trails are publicly available in a shared open access dataset [192].



(a)



(b)

Figure 4.6: The walking indoor (a), and outdoor (b) paths for examining the system performance were selected within the UCLouvain campus, and each of them contains the five locomotion modes. The numbers and arrows refer to the terrains order and direction over every path. Color codes correspond to those provided in Table 3.1.

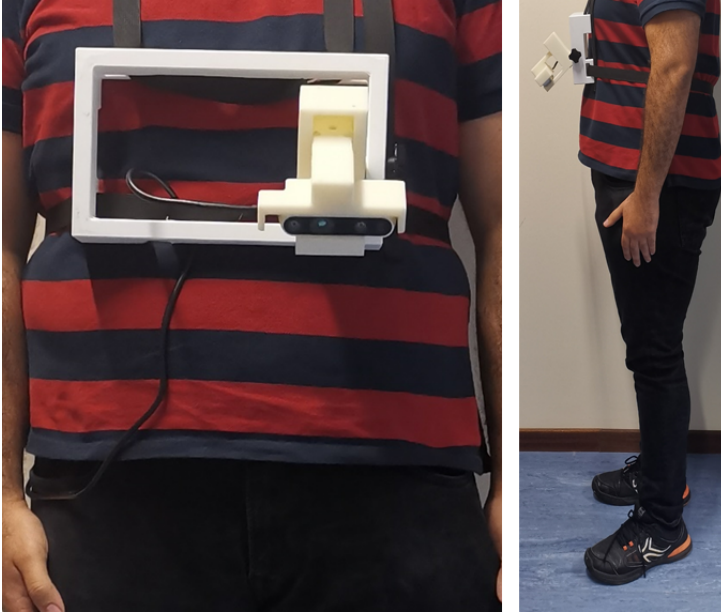


Figure 4.7: Wearable equipment to hold the depth camera on the user's chest

4.3.3 System Training and evaluation

In this part, we only relied on the first stage of the algorithm through processing the extracted features from the detected planes within the collected data (see Fig. 4.2). The features extraction step produced around 68k rows of features corresponding to as many detected planes that afford various locomotion tasks, i.e., about 1000 planes per trial. In order to estimate the error rate of the classification algorithm, we ran a cross-validation setup, as typically used when an independent set of testing samples is not available [193]. Indeed, it is one of the most popular data resampling methods for assessing the generalization ability of a predictive model while avoiding overfitting [194]. Accordingly, the data were split into 8 folds corresponding to the 8 participants, and 8 different models can be trained by keeping the folds of 7 participants as training data, and the remaining one as test data (see Fig. 4.8). The overall system performance was computed by averaging all the obtained accuracies. For selecting the classifier displaying the best performance, this process was repeated with several classifiers such as SVM with radial basis function kernel, Decision Tree (DT), QDA, LDA, and Naive Bayes (NB) classifiers [195].

Besides computing the achieved accuracy corresponding to each classifier, the required time for predicting the terrain type based on its extracted features was, also, calculated. The computational cost per classifier was computed by determining the time required for predicting the locomotion mode of a single plane. This was performed by feeding the trained model with a single row of features corresponding to that benchmark plane. It is considered to compute the time required to predict the locomotion modes corresponding to one million planes. The features of those planes are set as inputs to the trained classifiers, and the required time by each classifier

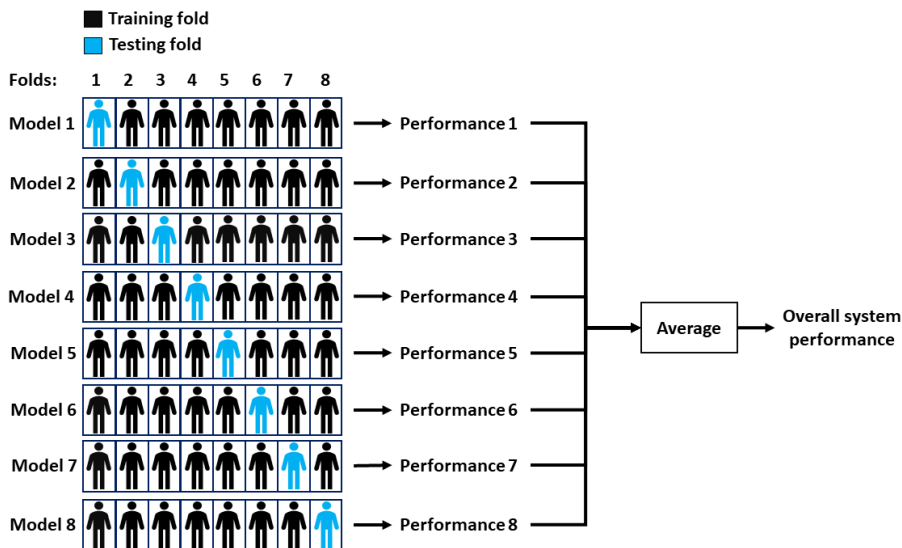


Figure 4.8: 8 folds cross-validation was performed by training each model based on the data collected for seven participants and testing it on the data of the remaining participant that was not included for building that model. The final result for five locomotion modes is obtained by averaging the accuracy obtained with the 8 trained models.

to achieve all the predictions is calculated. This process is repeated ten times to increase the resolution of the measurements, and the recorded computational costs are averaged.

4.3.4 System performance estimation for the three steps

After selecting the classifier with the best accuracy, each one of the 8 models was tested against the recorded trials that were not included in the training of that model, and the accuracy in detecting each of the five locomotion modes for three steps in advance was independently computed for each model in four conditions: indoor and outdoor; each with clear and partially occluded path as illustrated in Fig. 4.9. The terrain predictions for three steps were obtained by running both stages of the algorithm (see Fig. 4.2). The ground truth was manually determined using the RGB footage, and the accuracy percentage for a given locomotion mode was calculated as the number of true positive samples over the number of the total collected samples corresponding to that mode (see equations 3.2 and 3.3). The final accuracies in the four conditions are obtained by averaging the corresponding ones that are recorded for all 8 models.

On the other hand, the geometrical features estimated for each detected terrain were compared with manually measured values of ramp inclination and stair riser height. Then, the mean absolute error was obtained for the estimated geometrical such as step height and ramp inclination.

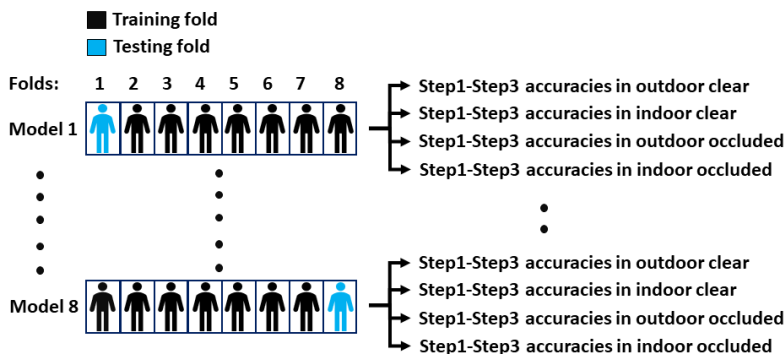


Figure 4.9: The locomotion modes accuracies for three steps in advance are obtained by running the two main stages of the algorithm for each trained model. The results are split according to the path and environmental conditions, and they are averaged for all models to obtain the final system performance.

Table 4.2: Classifiers Performance that is determined through applying cross validation over the 8 folds of the collected trials.

Classifier	G [%]	AS [%]	DS [%]	UR [%]	DR [%]	Grand Average [%]
SVM	95.9	97.97	97.54	96.15	96.28	96.8
DT	92.96	98	98.03	90.85	93.61	94.7
QDA	88.16	94.67	85.36	96.21	94.71	91.8
LDA	91.59	95.53	84.39	59.89	91.66	84.6
NB	93.19	95.85	67.93	96.29	90.75	88.8

Table 4.3: Average computational time needed to achieve the classification of one million planes given their corresponding input features.

Classifier	SVM	DT	QDA	LDA	NB
Averaged computational time [s]	182.81	0.14	0.09	0.37	0.34

4.4 Results

4.4.1 Classifiers performance

Extracted feature samples of the planes corresponding to a typical experimental trial are depicted in Fig. 4.10. The figure illustrates that the five locomotion modes indeed correspond to separated regions of the feature space. By applying cross-validation to the extracted features, various prediction accuracies are obtained with several classifiers, as listed in Table 4.2. The average processing time required by each classifier to perform 1M predictions are reported in Table 4.3. According to the results, LDA and NB achieve the lowest performance in comparison to the other classifiers, while a better performance is attained with the QDA classifier. The SVM and DT classifiers achieve high accuracy among the others for correctly predicting the terrain type

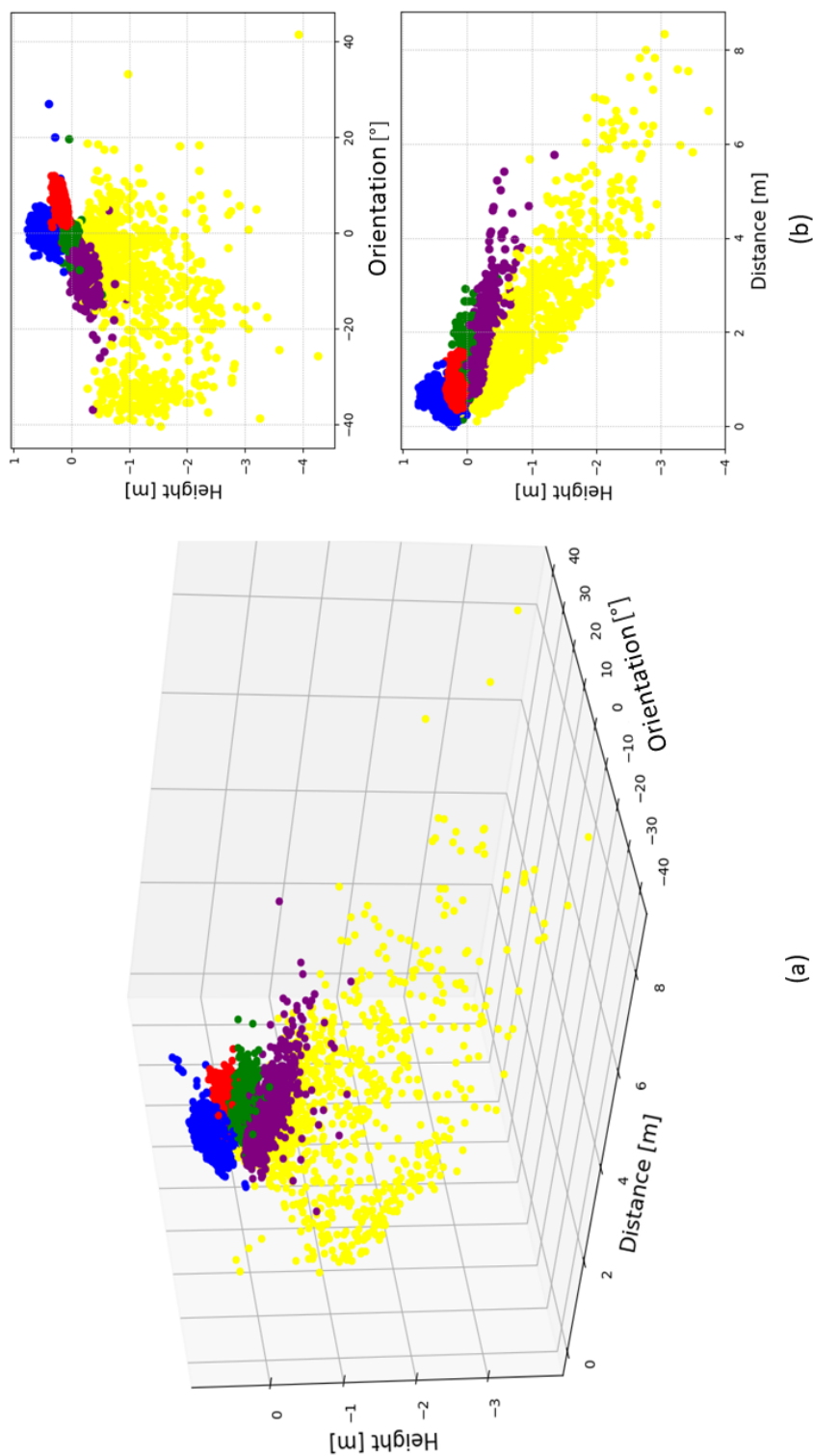


Figure 4.10: (a) Distribution of the extracted features Height (H_w), Distance (D_z) and Orientation (S_w) corresponding to the five locomotion modes. The color code is consistent with the one provided in Table 3.1. (b) 2D projections of the same data set, provided for visualization purposes.

based on the used features. Both classifiers reached similar accuracies for staircase modes (AS and DS). However, the SVM overcomes the DT in predicting the G, UR, and DR modes. Therefore, it was considered to be the main classifier of the presented algorithm. On the other hand, the results show that the SVM is the most computationally greedy among the other examined approaches which approximately have the same computational cost.

4.4.2 Locomotion-mode classification

Typical detection samples in clear path are provided in Fig. 4.11 with the color codes reported in Table 3.1. In general, the algorithm provided a good separation between the 5 different locomotion modes. In particular, the transition between the ground and ramp was well detected although there was no physical separation between them in the tested environments. Detection samples with partially occluded paths are reported in Fig. 4.12 with the same color code. This figure particularly illustrates how the walker in front of the participant occluded the scene.

The determined accuracy for the upcoming three steps is reported in Fig. 4.13 for the 4 experimental conditions¹ (indoor and outdoor with clear and partial occlusion for each). Moreover, to give a better insight into the wrong recognition results, we computed the confusion matrix for each of the three steps independently. They are provided in Fig. 4.14 (a) to (c) for clear path and in (d) to (e) for occluded paths.

With clear paths, the system achieved 98% as the grand average accuracy over all the locomotion modes at the first step, with more than 95% accuracy for each locomotion mode in indoor and outdoor environments. For the second step, the system displayed lower grand average accuracy (about 92%), but still above 90% for most of the locomotion modes. This accuracy dropped down for the third step, in particular for some locomotion modes. The decrease of accuracy was indeed particularly critical for the stair modes (both ascending and descending). In the case of partially occluded paths, the algorithm kept a high level of performance, at least for the first step in front of the participant and for all locomotion modes but the one corresponding to descending stairs (DS). In the DS mode, the occlusion indeed impacted the detection accuracy, which dropped to 71% for indoor and 84% for outdoor environments. For the second and third steps, the occlusion showed small to no effect on the ground and ramp terrain types, but again a stronger influence was observed for ascending and descending staircases, as reported in Fig. 4.13 (c) and (d). Confusion matrices reveal that no other locomotion mode was detected in these cases since most of the confusions were done with the ND label. This was expected since the occlusion was covering the camera's field of view, especially for the third step.

4.4.3 Terrains geometrical features

Samples of the estimated geometrical features compared to the measured values are provided in Fig. 4.15 for ground and ramp terrains, and in Fig. 4.16 for ascending

¹For the detection results recorded for each participant during locomotion over four paths with a clear and partially occluded condition, see Appendix B.

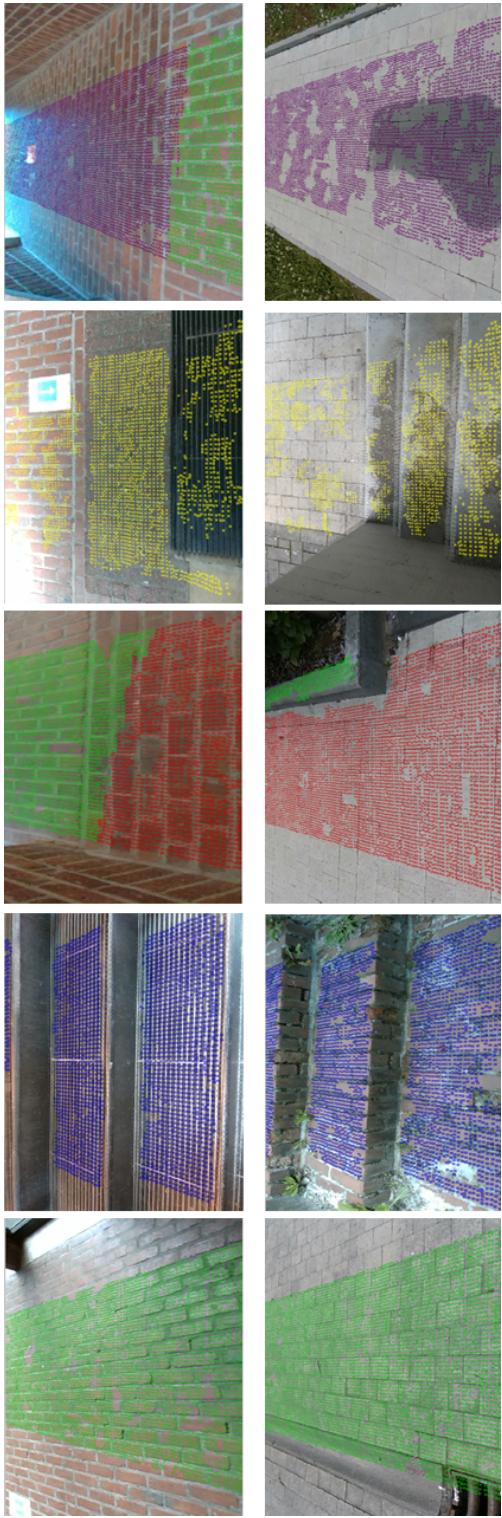


Figure 4.11: Detection samples of different terrains, overlaid on actual footages. The top row shows samples of detection in an indoor environment, and the bottom row reports detection samples in an outdoor environment. Color code is consistent with Table 3.1.

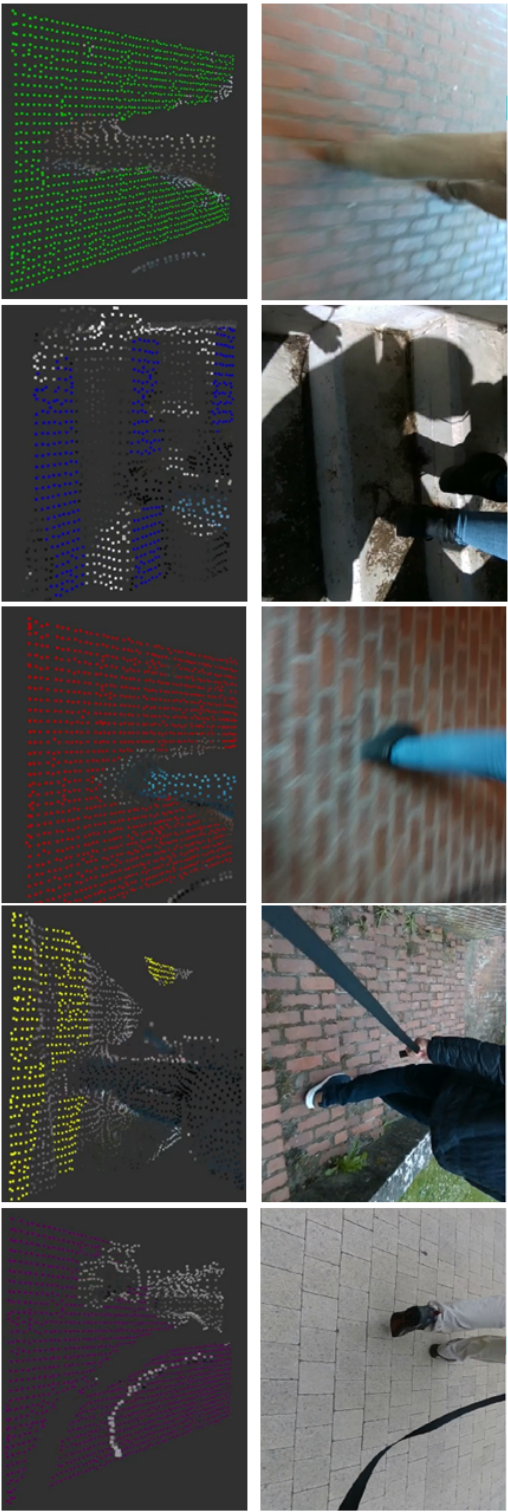


Figure 4.12: Detection samples of different terrains in partially occluded environments. The top row shows real RGB footages while the bottom one shows the algorithm detection results with points labeled with colors provided in Table 3.1. A 0.9 m rope was held from each side by the participant and the leading walker who was occluding the scene, to loosely keep a constant distance between them.

and descending staircases. Fig. 4.17 reports the mean absolute error of these features computed in clear and partially occluded conditions. It reveals that our system estimated the plane inclination during the flat ground and up/down ramp modes with limited error both with and without occlusion. The step height was also well estimated in the ascending stairs mode, with only a few centimeters of absolute error. The largest amount of absolute error was recorded for the descending stairs mode for both path conditions. This is consistent with Fig. 4.10, showing more variability for that locomotion mode.

4.4.4 Frame processing rate

Considering the intended application of locomotion assistance, the computational load and execution speed of the proposed algorithm are critical for real-time implementation. In our experiments, we observed that the frame processing rate was directly dependent on the size of the cloud to be processed, and thus indirectly on the locomotion mode itself. Indeed, the processed cloud typically contained fewer points for the ascending tasks (stairs and ramp) than for overground walking; which itself contained fewer points than descending tasks (stairs and ramp). Therefore, we observed that the frame processing rate was the best for ascending tasks – where it reached the maximum of 15 fps, corresponding to the publication rate of the camera. It dropped to about 14 fps for overground walking, 11 fps for down ramp, and 7 fps for descending stairs.

4.5 Discussion

It is not surprising that the SVM overcomes other tested classifiers in correctly identifying the locomotion modes of the detected planes, although it is relatively the worst regarding the computational cost. Indeed, these outcomes are in agreement with the conclusions of the review study reported in Chapter 2. Moreover, the distribution of the extracted features in the space (see Fig. 4.10) supports the approach based on SVM classification with radial basis function kernel because the data are not linearly separated. This, also, justifies the reason behind the lowest classification outcomes obtained by LDA and NB classifiers, and higher recorded accuracies with the other approaches that have non-linear decision boundaries.

In terms of overall system outcomes, the algorithm showed remarkable detection performance to identify various locomotion tasks, at least for two steps in advance with clear path and one step in advance with occluded path. In general, there was a drop in the detection accuracy as a function of the distance of the prediction step. This was due to the inherent noise of stereo-based depth measurement, which increases with the distance to objects. This drop was the most critical for the third step, since the accuracy decreased for all locomotion modes, and particularly for staircases. For ascending stairs, this was likely due to the fact that the third step became out of the field of view of the camera most of the time, as exemplified in Figs 4.11 and 4.12. For descending stairs, the camera was not able to identify the third step many times because of the alignment between the camera axis and the perspective of the stairs. Consequently, the distance between the third step and the camera was larger than

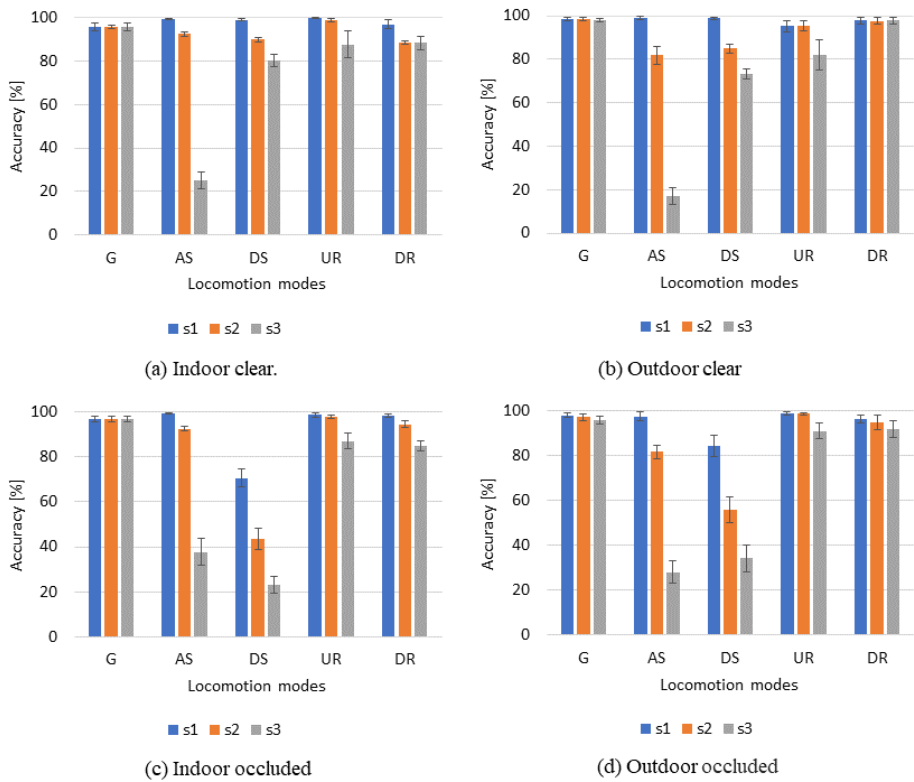


Figure 4.13: Accuracy of the detection of the 5 locomotion modes. Error bars represent standard errors across participants. They are independently reported for three steps in advance ahead of the walker, labeled with s1, s2, and s3. The four panels correspond to the four tested environmental conditions.

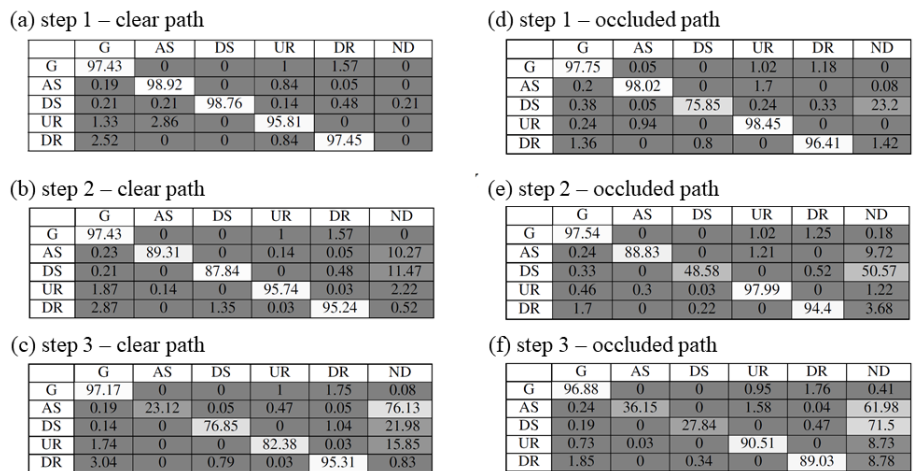


Figure 4.14: Confusion matrices of the detected types of terrains. Values are given in percents.

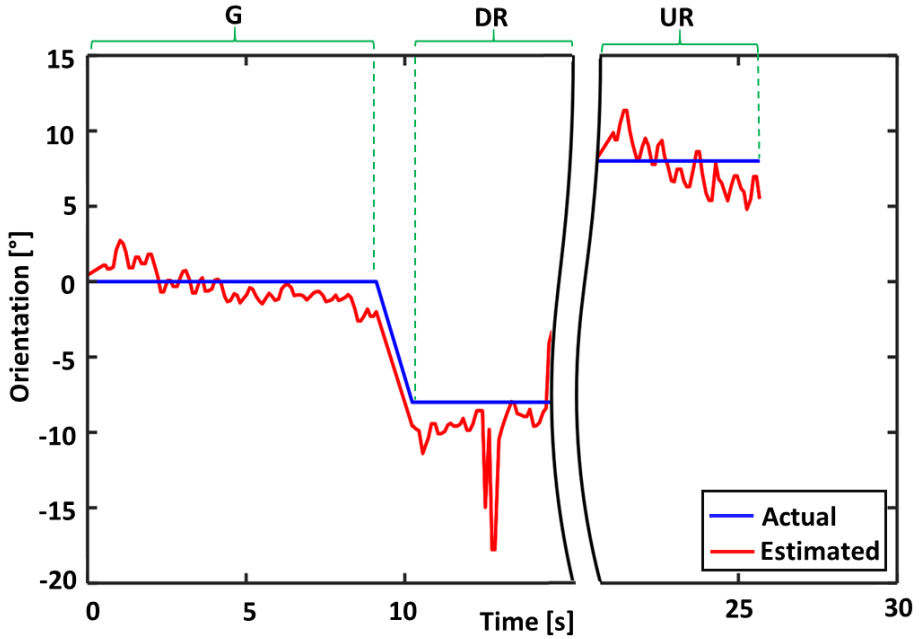


Figure 4.15: Estimated Ground inclination during walking on flat and inclined grounds (red curve). The blue curve displays the actual inclination measured on the field.

in the other conditions, and the plane fitting approach did not work anymore. Instead, the camera captured a distorted cloud, and the region growing algorithm did not work properly. The confusion matrix for this third step (Fig. 4.14(c)) when descending steps reveals that it was actually most of the time labeled as no detection (ND), and not confused with another locomotion mode.

Interestingly, the presented results did not reveal differences between indoor and outdoor environments, in all locomotion modes even with partial occlusion. This is due to the fact that we used a stereo camera technology which is working equally with artificial and ambient sunlight [188]. Therefore, we can confidentially claim that our system is robust to work in both environments with their respective lighting conditions. However, the rightmost sample of the outdoor footages in Fig. 4.11 displays many spots on the path which the camera was not able to capture. Interestingly, the algorithm was nevertheless able to reconstruct the affordance despite these gaps because it relies on a 3D-point cloud to fit a global plane, even if there are some discontinuities in the cloud.

Several depth vision based approaches were published for locomotion assistive devices [43, 44, 136, 148, 156]. In general, they were developed to predict the upcoming terrain type for a region of interest that may correspond to a single step in a clear path condition. Therefore, the comparison to our results should focus on those obtained for the first step in the clear path condition. In that case, our system recorded above 95% of accuracy for all locomotion modes, with a grand average about 98%. This is similar to the average accuracy reported in the most recent

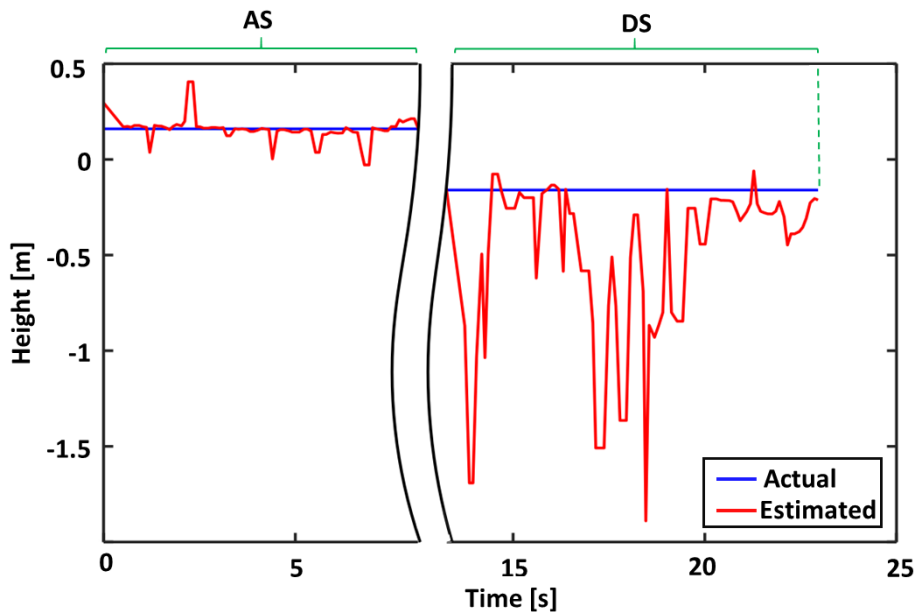


Figure 4.16: Estimated Step height for ascending and descending stairs (red curve). The blue curve displays the actual step height measured on the field.

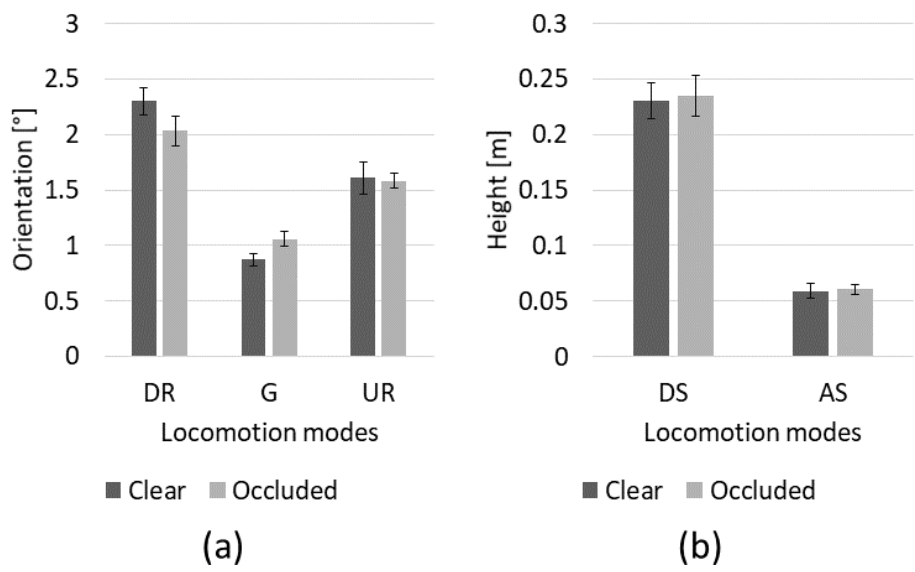


Figure 4.17: Absolute error of geometrical features estimation in clear and partially occluded path conditions. (a) Mean absolute error for ground inclination during flat ground and up/down ramp modes. (b) Mean absolute error for step height during ascending and descending stairs. Error bars represent standard error across participants.

papers, over the same 5 locomotion modes [43, 44, 196]. Moreover, our results might be considered as being more robust, since we implemented two different indoor and outdoor paths, while the former approaches tested only one. Finally, to the best of our knowledge, this is the first time that such system has been tested with partial occlusions caused by another walker. Our results showed that our approach is robust to that perturbation, at least regarding prediction accuracy for the first step ahead of the user.

Geometrical features estimation by our algorithm comes with low error for all terrain types, even with partial occlusion, except for the DS mode. This is visible for a typical trial in Fig. 4.15 and 4.16. It is also confirmed by overall statistics in Fig. 4.17 where the errors in estimating the geometrical features are occurring due to the noisy sensory readings (IMU and/or camera). For instance, during DS mode the camera orientation is aligned with the scene; this makes plane more difficult to detect due to fact they are more parallel to the axes of the camera which reflects negatively on the estimated geometrical features. Interestingly, descending tasks, in which our approach recorded higher error rates in estimating their geometrical features, are likely those for which precise feature estimation is the least critical. This is because they do not require feature-dependent active energy injection, and the device could operate as a globally dissipative system. In contrast, the AS mode does require accurately identifying the step height in order to inject the needed amount of energy for achieving a more natural gait pattern [197]. In comparison to other studies, competitive results for geometrical features estimation were reported by [36, 43]; however, they were not validated against partial path occlusion and/or in multiple indoor and outdoor paths.

The size of the recorded point cloud is generally subjected to the scene and the field of view, and this has a direct impact on the algorithm frame rate. In particular, the algorithm reached higher frame rates during overground and ascending tasks than during descending ones. In the latter, the camera axis is indeed more aligned with the environment perspective (ground or staircase inclination), thus corresponding to a larger intersection surface and generating more points in the cloud. Furthermore, it is possible that a wearable minicomputer with a more limited processing power would lead to a smaller frame processing rate. We speculate that the frame rate should not drop below 3 fps so that the high-level control strategy of the device could be updated about 3 times per step. Future work will determine if this target is achievable with a fully wearable computer.

Estimating the locomotion terrain type and its geometrical feature for several steps upfront of the user can lead to achieving better predictability and transition patterns for people equipped with active prostheses or exoskeletons. Indeed, these devices should behave differently as a function of the terrain, and these transitions should be anticipated in order to prepare the upcoming steps in the most appropriate way. People with vision disorders would also benefit from such a device. Indeed, it could provide augmented artificial feedback compensating the disabled vision, such as auditory and/or haptic cueing. This would help the disabled in anticipating transitions between locomotion modes, and thus contribute to overall stability.

4.6 Conclusion

In this chapter, we presented a wearable computer vision-based terrain detection system. The system processes the obtained points cloud from the depth camera to extract environmental features and achieve terrain detection by feeding these features into an SVM classifier. The detected affordances are further processed to estimate the terrain types for three upcoming steps. Our system achieves high accuracy (above 95%) for various terrains recognition one step ahead of the user, and with a clear path. Also, it displays a good ability to recognize different kinds of terrains even when they were partially occluded. We further tested the ability of the system to achieve a similar detection for two more steps in front of the user. Results stayed consistent for the second step and got degraded for the third step, at least for the ascending and descending staircase modes, and/or with occlusion.

The step size in Section 4.2.2.a was assumed to be fixed and equals 0.5 meter for all trials. However, a human selects his/her stride length based on several factors such as walking speed, footwear, injury history, age, etc. [198]. This stride length, consequently, varies from one person to another. Equipping the terrain detection algorithm with the step size estimation capability will rise its ability to accurately matching between the terrain type estimated for a specific step and the user's foot placement on that step. In addition to the step size estimation, estimating other parameters such as walking speed, and cadence can provide a good tool for the design engineer to track the assistive device performance over several terrains. Detecting turning angle could be further beneficial to adjust the assistive device stiffness [199]. Accordingly, the next chapter provides speed, cadence, step size, and turning angle estimation approach by analyzing data collected by an IMU attached to the depth camera.

Walking speed, cadence, step size and turning estimation

5

5.1 Introduction

Over the past decade, estimation of human walking speed, step size and direction are a topic of interest in many research areas such as clinical studies, sports training, gait analysis, and pedestrian localization [200, 201]. Visual inertial odometry (VIO) can be considered as a potential method for computing the speed, turning angle and the step sizes. Indeed, it demonstrates a good ability for human localization in indoor environment [202]. Even though, it can be influenced by the environmental conditions such as texture [203]. For instance, in case of low-texture scenes, it becomes challenging to extract several features and track them from one frame to another [203]. Particularly in our case, the camera is pointing to the ground which makes it more vulnerable to the low-texture scenes, and it could also possibly fail in case the scene was occluded with a walking person in front of the user. Therefore, we will rely on user dependent sensor to obtain such information.

In this chapter, we present several approaches for estimating the step size and turning angle by benefiting from the built-in IMU sensor of the depth camera attached to the user's chest. The former is fed into the terrain detection system for achieving terrain recognition for multiple steps with adaptive stride length, and the latter can be used to detect the user turning in order to enact a change in stiffness during the swing phase of gait while the prosthesis is unloaded [199]. We compute the step size by estimating the walking speed and cadence. The following sections elaborate on the approaches applied for walking speed, cadence, step size, and turning estimation. Then, we introduce the experimental methods for assessing the performance of our algorithms. Afterward, the achieved results are presented and discussed. The chapter ends with a conclusion.

5.2 Walking speed estimation

Walking speed is an essential factor in the biomechanical related analysis and general locomotion research [204]. Many environmental and wearable technologies are

developed recently for walking speed estimation. Camera-based tracking, floor vibration, and walkway are several instances of environmental or non-wearable technologies for walking speed estimation [205, 206]. These systems are expensive and they are restricted in a laboratory environment setup [200], while it is challenging to have them implemented in real-life [205]. Consequently, wearable technologies are more feasible to be implemented in an uncontrolled environment [205]. IMU is the most popular wearable sensor for walking speed estimation [205]. Recent technologies mount single or multiple IMUs sensors on the human body for detecting the movement of the body, estimating the energy expenditure, step size, and speed which turns out to be a promising technique due to the small size of these sensors. Also, they are inexpensive, and they can afford comfortable sensor configuration to the users [201]. Moreover, these systems can be examined in various environments without the need for a special setup.

Several sensor configurations and algorithms are implemented to achieve speed estimation with IMU sensor(s). Some methodologies rely on mounting multiple IMU sensors on the human body [207]. For instance, in [208], the authors used three IMU sensors mounted on the thigh, shank, and foot of the subject. Then, the collected data are processed and fed to a neural network to estimate the walking speed. However, we are more interested in single IMU-based methodologies since they are more relevant to our application.

A single IMU placement on the human body is used in many approaches with various configurations. For instance, IMU placed on the waist is applied in several approaches for speed estimation through incorporating a biomechanical model of human motion such as the inverted pendulum model [209–211]. Other articles use inertial-based approaches for speed estimation. These methods rely on integrating the linear acceleration measurements obtained from IMU sensors placed on one of the wearer's feet to compute the speed [200, 212, 213]. However, direct integration of the raw IMU-accelerometer signal suffers from the growth of unbounded errors, i.e. drift that is caused by the presence of measurement noise and sensor bias [214]. Machine learning regression models are incorporated in several methodologies to estimate the walking speed. These methods extract multiple features from the IMU signal(s) and feed them into regression module to achieve speed estimation [201, 215–218]. Unlike the previous approaches, the machine learning based methods are introduced in the literature with several sensor configurations such as chest, head and hip [201, 215–218]. Therefore, such algorithms draw our attention since they are applied to sensor configuration that is close or similar to our set up [216, 218]. However, they suffer from delay which can be more than five seconds in obtaining the walking speed which can cause an issue in our application that needs an instantaneous walking speed estimation with minimum delay. Therefore, our target is to implement a machine learning based walking speed estimation algorithm that estimates the instantaneous speed in a relatively short period of time to be used later on for predicting the walking step size. In the following paragraphs, we describe the implementation of our speed estimation algorithm.

Our algorithm relies on exploiting the triaxial accelerometer data obtained from the built-in 6-DOF IMU of Realsense (D435i) depth camera. Moreover, we computed the acceleration norm which is defined as:

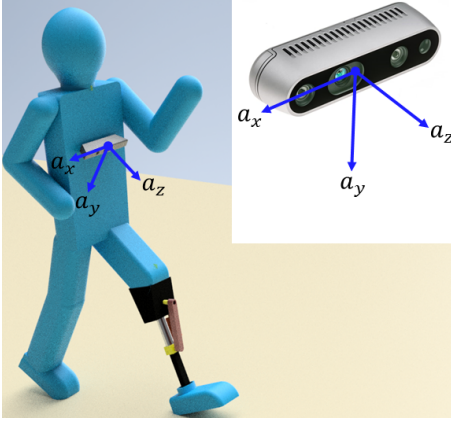


Figure 5.1: The acceleration components of accelerometer which belongs to the built in IMU sensor of the depth camera. According to [220], these component are directed with respect to the camera orientation frame

$$a_n = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (5.1)$$

where a_x, a_y and a_z are the components of acceleration in x, y and z-directions with respect to the camera frame (see Fig. 5.1). Computing the acceleration norm for the speed estimation implies insensitivity to the sensor placement and orientation [219].

The computed norm is, then, fed into several processing steps to achieve speed estimation as shown in Fig. 5.2. Based on several trials, 1 sec length sliding window is selected with 50% overlap with the previous window samples, and since the accelerometer of Realsense depth camera publish rate is 200Hz, the window gathers 200 samples each half a second due to the 50% overlap. The collected samples are passed to a Butterworth lowpass filter to be filtered at 5Hz cutoff frequency in order to get rid of noise [216]. Then, we extract several features from the filtered waveform in time and frequency domains. Time domain features are zero crossing, mean, signal magnitude area, maximum and minimum. In the frequency domain, we pick the magnitude of the first four frequency coefficients of the signal spectrum obtained after applying fast Fourier Transform (FFT). In total, we thus extract nine features that are stored in a feature vector passed to a trained regression model to achieve speed estimation. As a regression model to predict the walking speed, we use Gaussian Process Regression (GPR) has already been used in previous work for similar purposes [215, 216, 221]. Indeed, this model is more popular to solve regression problems than other methods such as Support Vector Regression (SVR) since it is a less ‘parametric’ tool that provides the estimation based on the data rather than a fixed model. Moreover, GPR showed superior performance when used for walking speed estimation in comparison to Least Square Regression model (LSR) and Bayesian Linear Regression (BLR) [215, 221].

5.3 Cadence and Step size estimation

In order to have the step size estimated, we rely on the following relation between the walking speed, cadence and step size:

$$Stepsize = \frac{WalkingSpeed}{Cadence} \quad (5.2)$$

while the speed is already estimated in meters per second, the cadence represents the steps frequency per second. Although the computed norm of the IMU placed on the human body can reflect the walking cycle during locomotion, it can be influenced by vibrations and noise. Therefore, the period is computed by relying on a lowpass filtered norm signal (see Fig. 5.2), and by determining the number of samples in the interval between two consecutive positive or negative peaks. However, a threshold is applied to the distance between the consecutive peaks in order to obtain a cadence value that copes with the maximum heuristic walking cadence (130 steps/min) reported in [222, 223], so if the first peak is detected, the algorithm seeks for the next peak after 100 samples. After computing the period, the cadence is obtained based on the following equation:

$$Cadence = \frac{f_s}{Period} \quad (5.3)$$

where f_s represents the sampling frequency, which is 200Hz in our case.

5.4 Turning angle estimation

By exploiting the IMU data, we used the quaternion to determine the rotation angles of the sensor, which are roll, pitch, and yaw as illustrated in Fig. 5.3. The yaw represents the rotation of the camera around its y-axis. Therefore, it roughly captures the user walking direction from 0° to 360° , since it is tilted down a bit. In order to

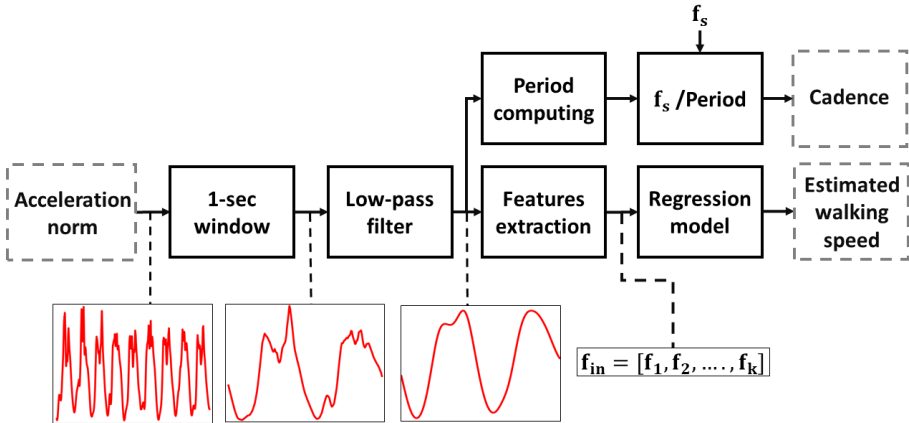


Figure 5.2: This block diagram shows speed and cadence estimation algorithms. The input is tri-axial acceleration norm signal computed from the accelerometer axis while the output is the estimated walking speed and cadence. f_{in} is the input features vector to the regression model to achieve speed estimation while f_s is the sampling frequency, i.e. 200Hz

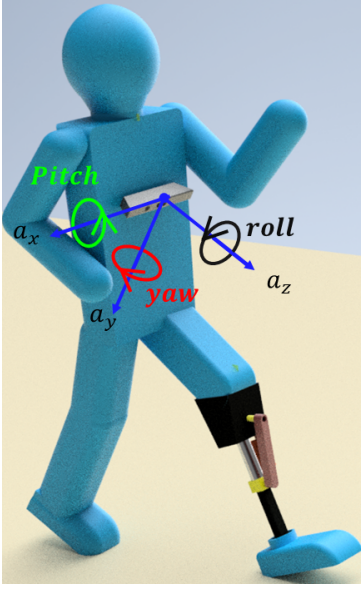


Figure 5.3: The rotation angles (roll, pitch and yaw) of the IMU sensor which is built in the depth camera. The turning angle T_{angle} is computed through determining the amount of change in the yaw angle over specific time window.

estimate the turning angle (T_{angle}), we rely on computing the change in the yaw angle over an interval of time according to the following equation:

$$T_{angle} = Y_{r1} - Y_{r2}, \quad (5.4)$$

where Y_{r1} represents the first yaw reading which is taken at time t , and Y_{r2} represents the second yaw reading which is taken at $t + dt$. In our case, dt equals to 0.5s. The magnitude of T_{angle} refers to the amount of turning while the sign of T_{angle} indicates the turning direction to the left or right.

5.5 Experimental methods

The implemented algorithms are, then, evaluated by conducting several trials with various paths. During each trial, we use the same wearable tool and camera introduced in Chapter 4 while the IMU-data and RGB images are the recorded data. The IMU data are processed later on for estimating the targeted parameters. The purpose of the recorded RGB images is for labeling the recorded IMU data and tracking the start and end of each trial. The following sections explore the applied methods to evaluate the walking speed, turning angle, and step size estimation algorithms.

5.5.1 Walking Speed and turning angle evaluation

Six participants (4 males, 2 females) aged (24-30) years are recruited for evaluating the speed and turning angle estimation approaches. For the same purpose, three paths are assigned as illustrated in Fig. 5.4. We name the three paths to be A, B, and C for the sack of convenience.

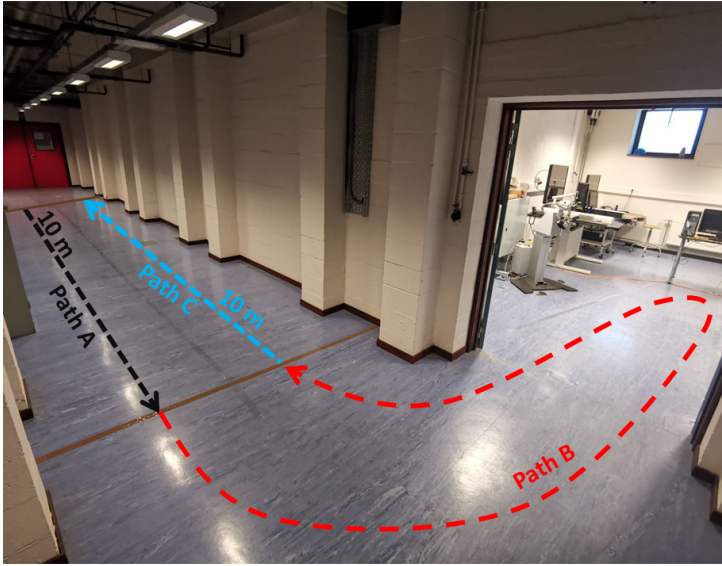


Figure 5.4: The allocated three paths (A, B and C) for evaluating the walking speed and turning angle estimation algorithms. Path A and C are used for testing and training the speed estimation approach, while the A, B and C paths are designed for turning angle approach evaluation.

In order to train the speed estimation regression model, we asked each participant to walk along path A, which is 10 meters long, at three different speeds: slow, normal, and fast. The collected data are labeled with the computed speed using stopwatch walking speed measuring technology [205] which is a popular method applied widely in clinical trials. It determines the speed by measuring the time a person takes to walk on a specific length path [205]. Totally, 18 trials are recorded and used to train the regression model.

Afterward, a new set of experiments is conducted by asking the six participants to walk at the normal walking speed on the three paths starting with A then B, and finally C (see Fig. 5.4). We use the recorded data for the three paths to evaluate the angle estimation algorithm while only the data collected over paths A and C are used to test the speed estimation algorithm.

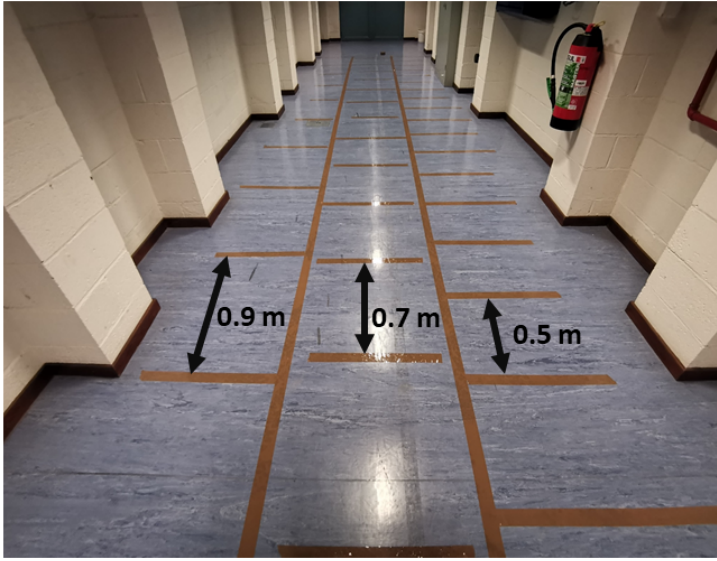


Figure 5.5: The three paths shown are designed for evaluating the step size estimation algorithm. Each path is marked with equally spaced markers, and by asking each participants to place his/her foot on the markers, we ensure of achieving a locomotion with fixed step sizes or lengths.

5.5.2 Step size evaluation

To test our step size estimation approach, we prepared three paths with three step sizes (0.5, 0.7, and 0.9 meters) that are close to the mean walking step lengths reported in [224,225], which are 0.79 m for men and 0.66 m for women. The paths are marked with tapes with fixed spacing for each path as shown in Fig. 5.5. Then, we asked three participants (two males and one female) aged between 24 to 30 years to walk along each path and step on the markers to achieve locomotion with a fixed step size. Therefore, we collected 9 trials in total. Afterward, the trials are labeled with assigned step sizes for each path and passed to the step size estimation method which depends on the estimated speed and cadence.

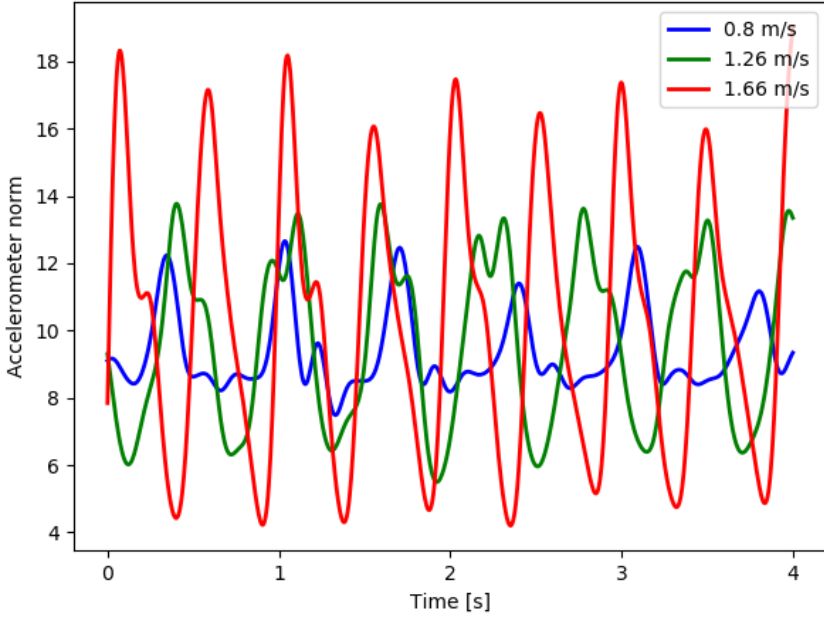


Figure 5.6: Typical filtered norm signals at three different walking speeds which are determined using stopwatch technology.

5.6 Results and discussion

In this part of the chapter, we explore and discuss the obtained results for the three implemented approaches: walking speed, step size, and turning angle estimation.

5.6.1 Walking speed estimation

Typical filtered norm signals for three different speeds in a duration of 4 seconds are displayed in Fig. 5.6, while the extracted time and frequency domain features out of these three signals are displayed in Fig. 5.7. We can observe a promising amount of separation among the elected features at the three different speeds. The features are extracted with a sliding window with 50% overlap which allows a walking speed estimation at each half a second. The estimated walking speeds on paths A and C are displayed in Figures 5.8 and 5.9 for the six participants (P1-P6). The estimated speeds are displayed alongside the calculated speed with stopwatch technology for each trial. According to the results, the system demonstrates the ability to predict a walking speed with Mean Absolute Error (MAE) equals to 7.7% in comparison to the calculated speed with the stopwatch method. The system average accuracy is also computed relying on the following equations which are given in [215]:

$$R_{err} = \frac{|estimated - actual|}{actual} \quad (5.5)$$

$$A_v(\%) = 100 - \overline{R_{err}}(\%) \quad (5.6)$$

Where R_{err} is the relative error while A_v is the average accuracy. By considering the speed computed using the stopwatch technology as the actual speed, the determined average accuracy is found to be equal to 93.5%. However, it is not guaranteed that each participant maintains a constant walking speed along the whole path, so we cannot totally rely on the computed walking speed with the stopwatch method to be the actual speed for the whole path. Therefore, an alternative and more concurrent approach is adopted to accurately determine the walking speed estimation algorithm performance. We compute the traveled distances for each trial along paths A and C which are 10 meters long each. The distance is calculated by integrating the recorded speeds over time for each trial and compared with the actual path length. The calculated distances for path A and C trials are shown in Fig. 5.10 and 5.11 respectively. The estimated distances show promising results with very close estimation of the actual path lengths. Indeed, the determined average accuracy is equal to 95.1% for the traveled distance estimation in a comparison with actual path length which implies an accurate speed estimation.

Moreover, our approach achieves accuracy degradation results that are close to similar implementations that use the same regression model (GPR) [215, 216] while an Artificial Neural Network (ANN) based one slightly overcomes our system by 2% [201] with average accuracy. However, these implementations suffer from several drawbacks. For instance, physical information about participants' bodies (e.g. weight and height) are utilized as anthropomorphic features and fed to the regression model. Such methodology may drive the regression model to produce biased predictions if it is tested with a different set of participants. Also, some implementations use relatively large sizes of sliding window (5 and 10 seconds) [215, 216] which cause several seconds of delay in predicting the walking speed while our implementation estimates the instantaneous walking speed with only half a second of delay and similar accuracy which makes it more reliable to be applied in several applications such as pedestrians localization, activity tracking, and walking assistive devices evaluation. Finally, a typical drawback of our system is its sensitivity to the IMU sensor movements. Therefore, any backlash in the sensor belt that mounts it over the user's chest can result in extra movement of the sensor which is interpreted by the system as parasitic movements.

5.6.2 Cadence and step size estimation

The cadence and step size results are displayed in Fig. 5.12 for the three participants (P1-P3), for the three paths with step sizes of 0.5m, 0.7m, and 0.9m. The detected cadence is in agreement with the observed values of people aged between 21 and 40 reported in [222], during moderate and vigorous ambulation. However, a single extreme outlier has been recorded during the trials at the end of the data of participant P3. This led to an estimated step size being larger than 1.5m, which is not physically possible. Therefore, a threshold is needed to filter out such an unusual outcome. The step size is published at the same rate of speed and cadence which is 2Hz. According to the results, the system achieves MAE of 8% and an average relative error of 11%. In comparison to literature, most of the implementations use different sensor configurations that mount the IMU on the foot or shank for stride

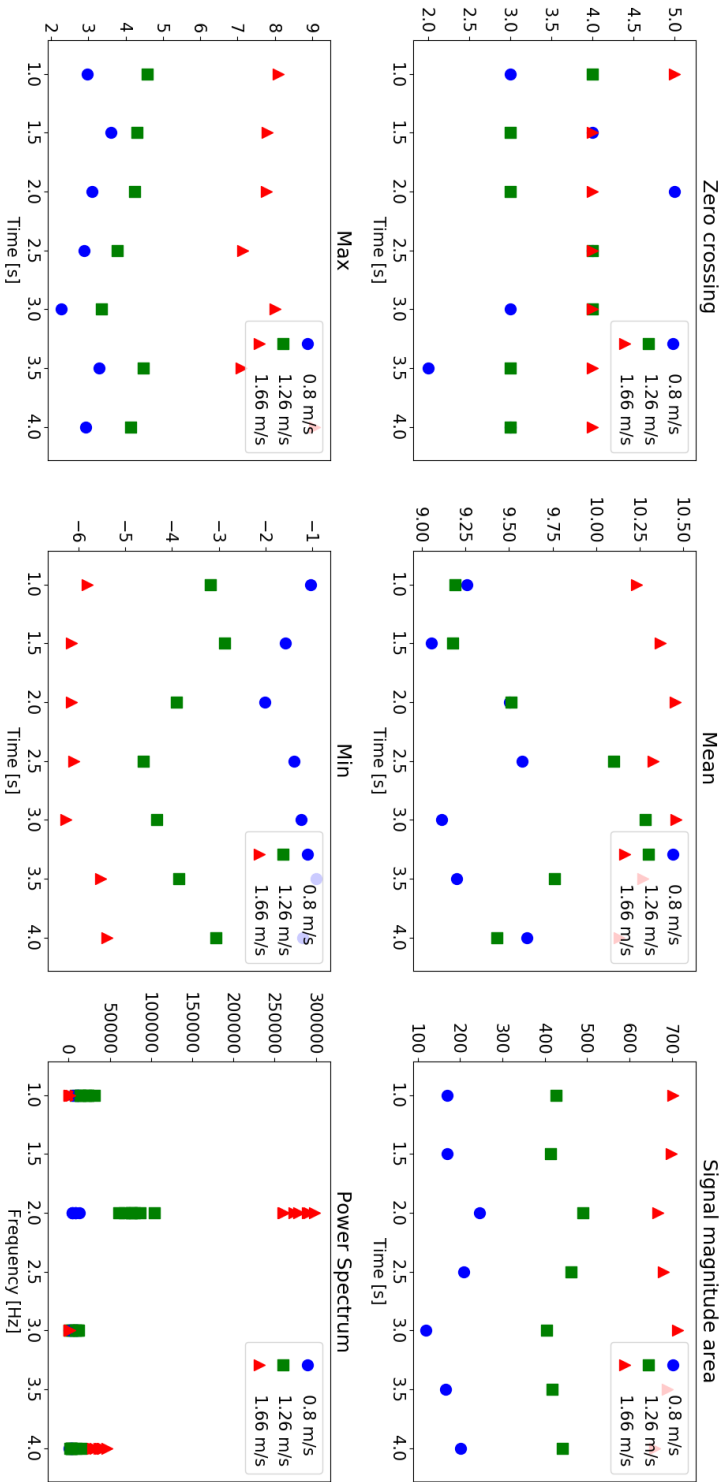


Figure 5.7: The extracted time and frequency domains features of the acceleration norm signals given in Fig. 5.6 for three different speeds. The features are extracted using sliding window of size of one second and with 50% overlap. Therefore, for a 4 seconds length input signal 7 features are extracted. Such features are used to predict the walking speed through feeding the to GPR regression model.

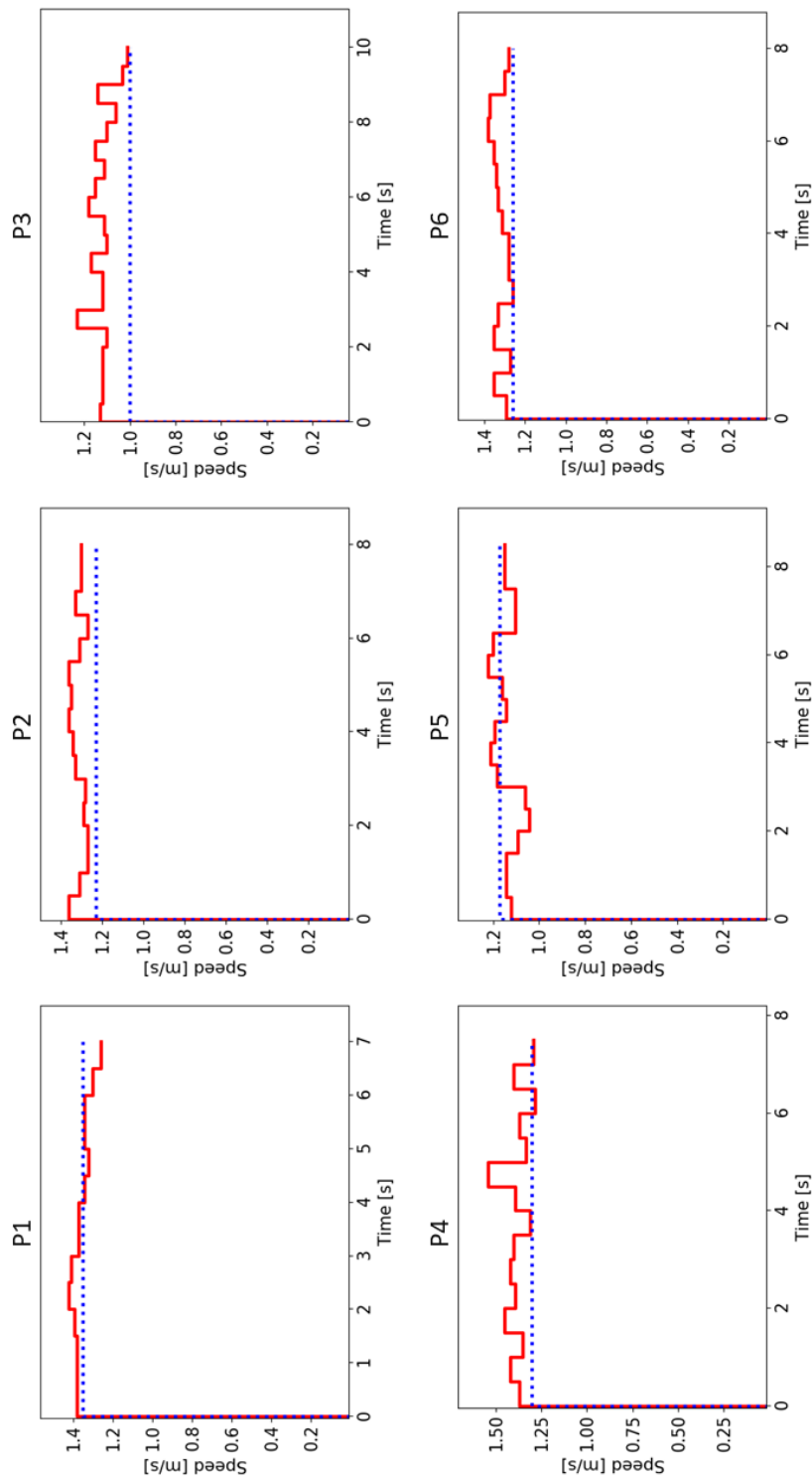


Figure 5.8: Predicted speeds of the six participants (P1-P6) during Path A trials. The dotted blue line represents the calculated speed using stopwatch technology while the red continuous line represents the estimated walking speed with our GPR based algorithm.

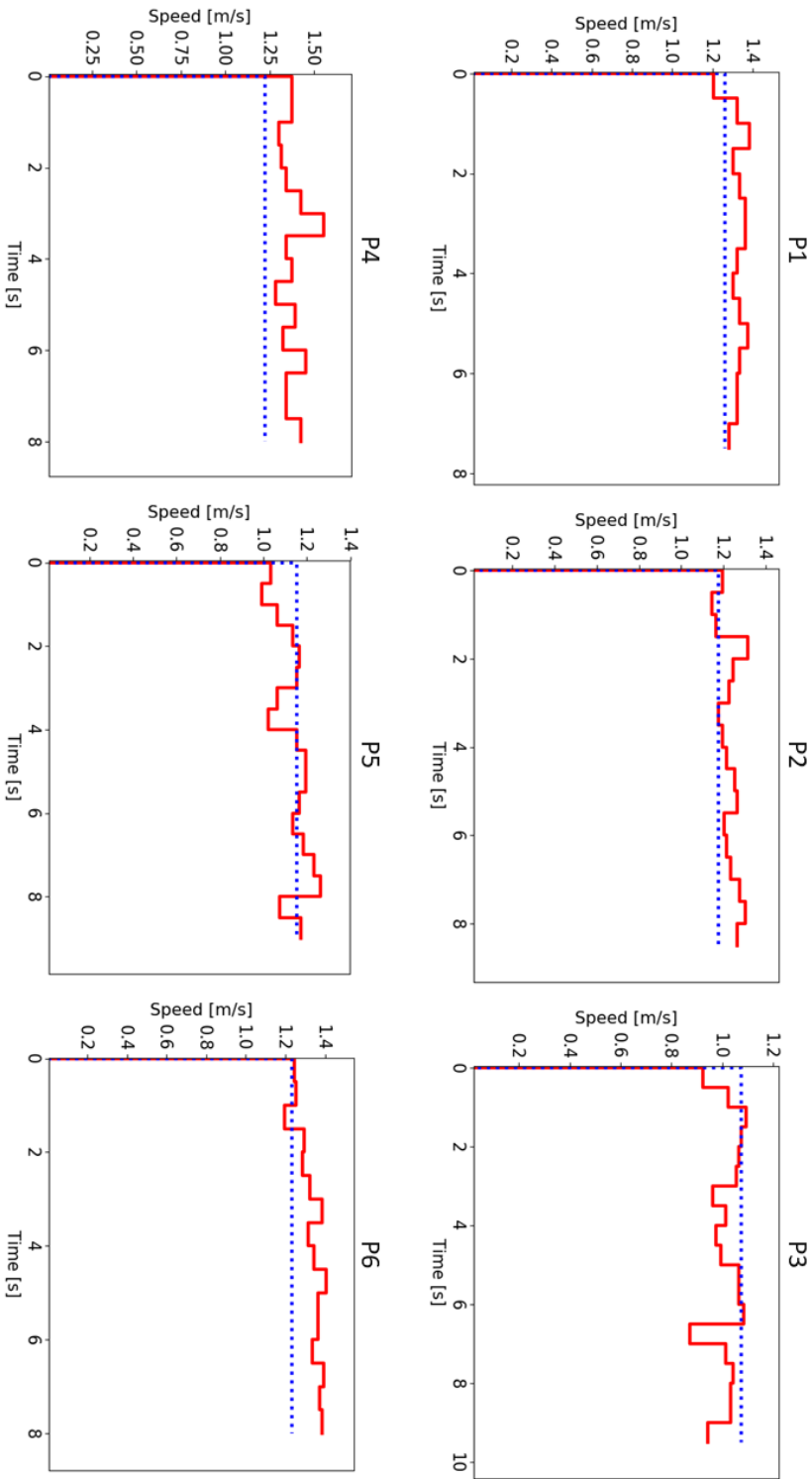


Figure 5.9: Predicted speeds of the six participants (P1-P6) during Path C trials. The dotted blue line represents the calculated speed using stopwatch technology while the red continuous line represents the estimated walking speed with our GPR based algorithm

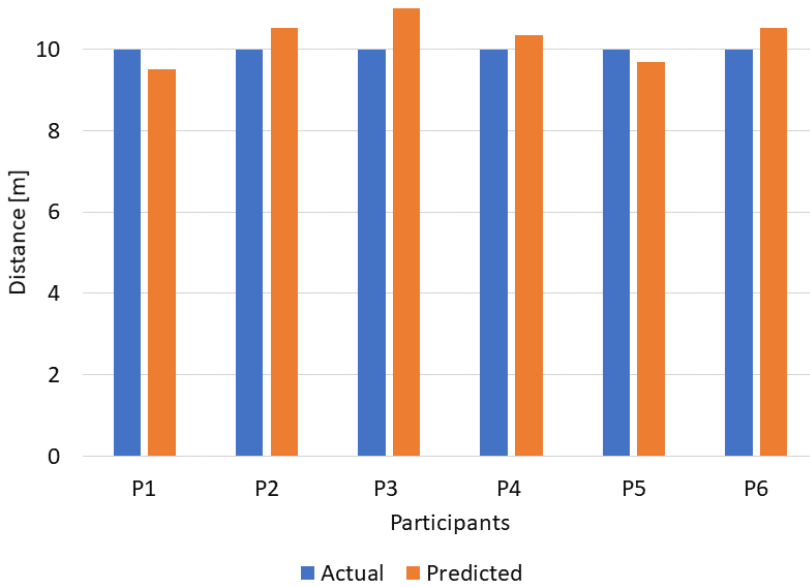


Figure 5.10: The calculated traveled distance for the six participants (P1-P6) based on integrating the walking speeds estimations for path A trials. The results are displayed alongside the actual path length which is 10 meters.

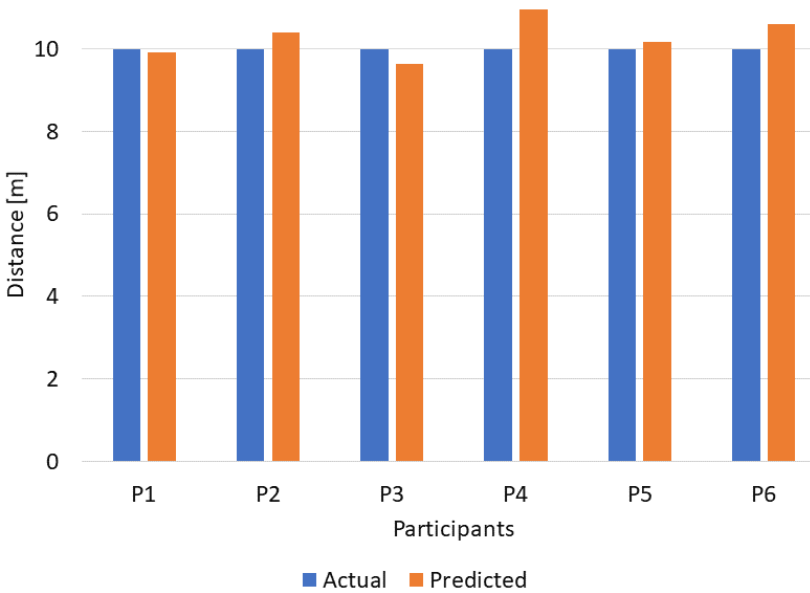


Figure 5.11: The calculated traveled distance for the six participants (P1-P6) based on integrating the walking speeds estimations for path C trials. The results are displayed alongside the actual path length which is 10 meters.

length estimation [226, 227]. However, recent approaches use chest-mounted IMU for indoor positioning [228] that achieves 10% error on step size estimation which is close to our rate. Therefore, our system demonstrates the ability to predict the instantaneous step size with an acceptable error rate that could cope with recently published implementation that uses a similar sensor configuration.

5.6.3 Turning angle estimation

The recorded data for the trials of the full path (A, B, and C) are used to evaluate the turning angle estimation approach. By incorporating the predicted walking speeds and angles, the locomotion trajectories during the three paths trials are computed through applying dead-reckoning method which is the process of computing one's position through estimating the direction and the traveled distance. The produced results for the six participants (P1-P6) are displayed in Fig. 5.13. According to the results, we can observe some deviations out of the actual path due to the accumulated errors in the speed and angle estimation. However, the major turns with 90° and 180° are detected accurately. Therefore, it is possible to rely on such an approach to be used for adjusting the control parameters of the assistive device during major turnings.

5.7 Conclusion

In this chapter, we presented walking speed, step size, and turning angle estimation algorithms. The approaches are evaluated through conducting several experiments with multiple participants. The speed prediction approach achieved more than 95.1% accuracy while the step size estimation algorithm was able to predict the step size with MAE equals to 8%. Afterward, the predicted speeds and turning angle are incorporated together using dead-reckoning method to localize each participant during his or her trial. Such methodology gave us a good intuition about turning angle estimation performance which showed a remarkable ability in capturing the major turns. Moreover, the implemented approaches were able to provide an instantaneous estimation of the targeted parameters at 2Hz rate. Therefore, such approaches can be useful for indoor pedestrian localization or locomotion assistive devices that need real-time updates.

In the upcoming chapter, we will combine the presented algorithms of this chapter with the terrain detection algorithm introduced in Chapter 4 in order to accomplish terrain detection that is adaptive to the user's stride length with the ability to predict the walking speed and turning angle.

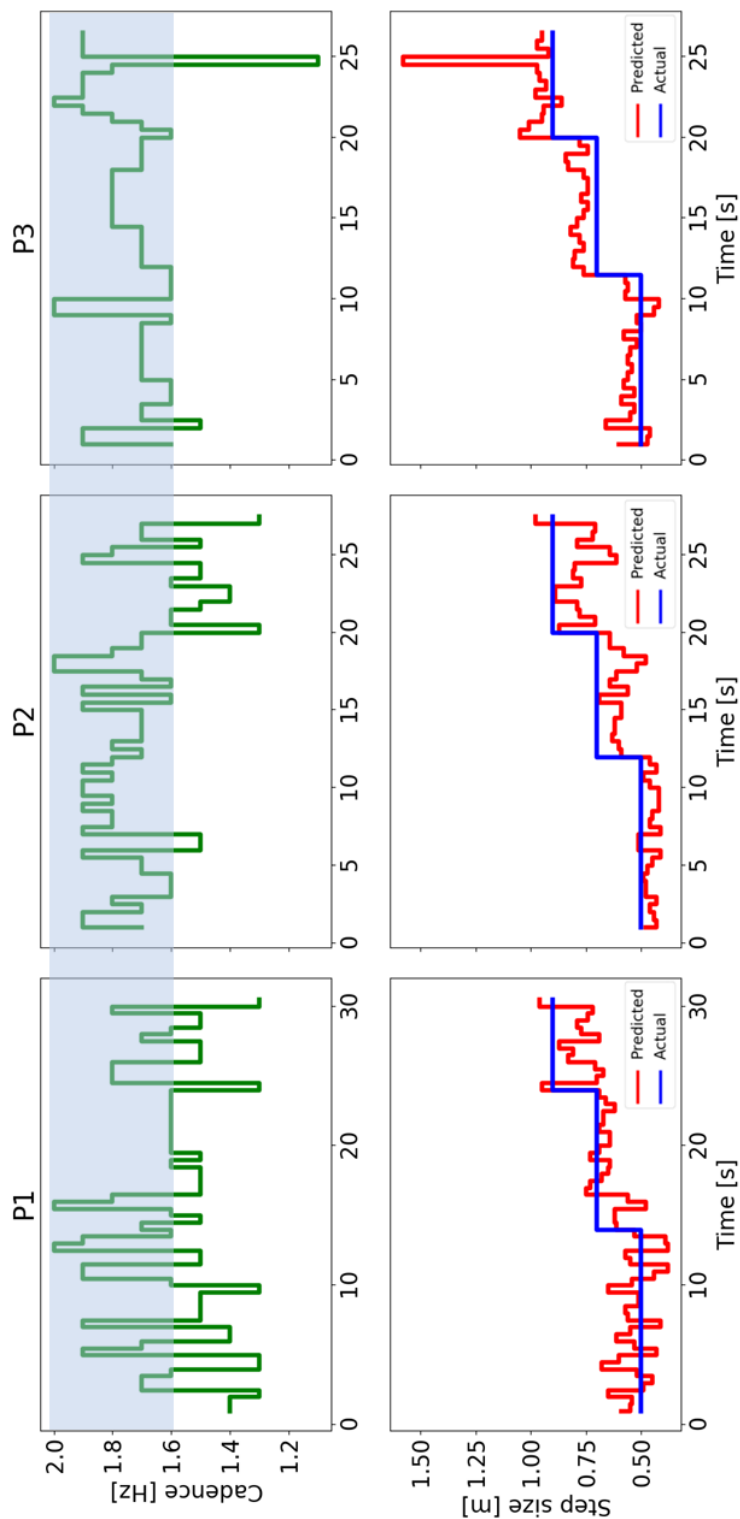


Figure 5.12: Estimated cadence (top panels) and step size (bottom panels) for the three participants (P1-P3). Each participant walked with three different steps sizes (0.5m, 0.7m and 0.9m). The shaded area represents the observed cadence values between moderate and vigorous ambulatory intensity in people aged between 21 and 40 years, according to [222].

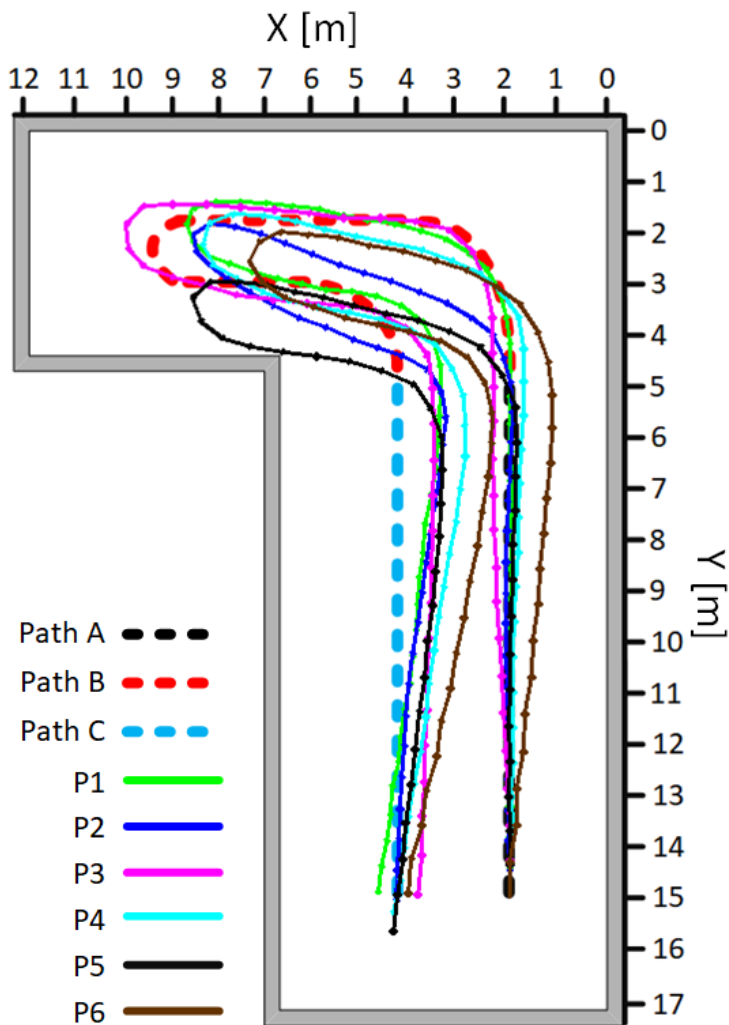


Figure 5.13: The dead-reckoning results for the six participants (P1-P6) computed through incorporating the predicted speeds and turning angles results when each participant walked on the paths A, B and C (see Fig. 5.4). The actual followed trajectory for the three paths are shown in dotted lines while the estimated trajectories are displayed in continuous lines.

Terrain Detection System for multiple steps with adaptive step size

6

6.1 Introduction

By having the step size estimation algorithm completed and validated, it is possible now to have the final version of our terrain detection system implemented through combining the algorithms developed in Chapters 4 and 5. Therefore, this chapter highlights several aspects of the obtained system with the focus on the followed methodology to determine the region of interest (ROI) for different step sizes which is not covered in detail in Chapter 4 since the step size parameter was set to be fixed. Also, we explore the possible outputs of the achieved system through preliminary results, and how they can be used to track the subject's activities over several kinds of terrain. Additionally, the prediction time computation of the system is introduced. The Chapter finally ends up with a conclusion.

6.2 System overview

A complete block diagram of the obtained system after linking the step size estimation approach is shown in Fig. 6.1. While the points cloud data are used for terrain recognition, the IMU data are heavily exploited to obtain the camera orientation, turning angle, walking speed, and step size. As outputs, the system provides terrain estimation for three upcoming steps with their geometrical features. Moreover, speed and turning angle are set to be outputs as well.

As illustrated in the block diagram, the estimation of the terrain for three steps process gets the instantaneous walking step size from the step size estimation process in order to achieve terrain detection that is adaptive to the user's stride length during ground and ramp locomotion modes while for the staircase an alternative methodology is applied (see 4.2.2.b). Since we apply the features transformation method (see 4.2.1.d) rather than the whole cloud transformation for saving computational cost, the obtained points cloud with affordances detection are still in the camera frame. Therefore in order to obtain the points cloud that belongs to upcoming steps, the

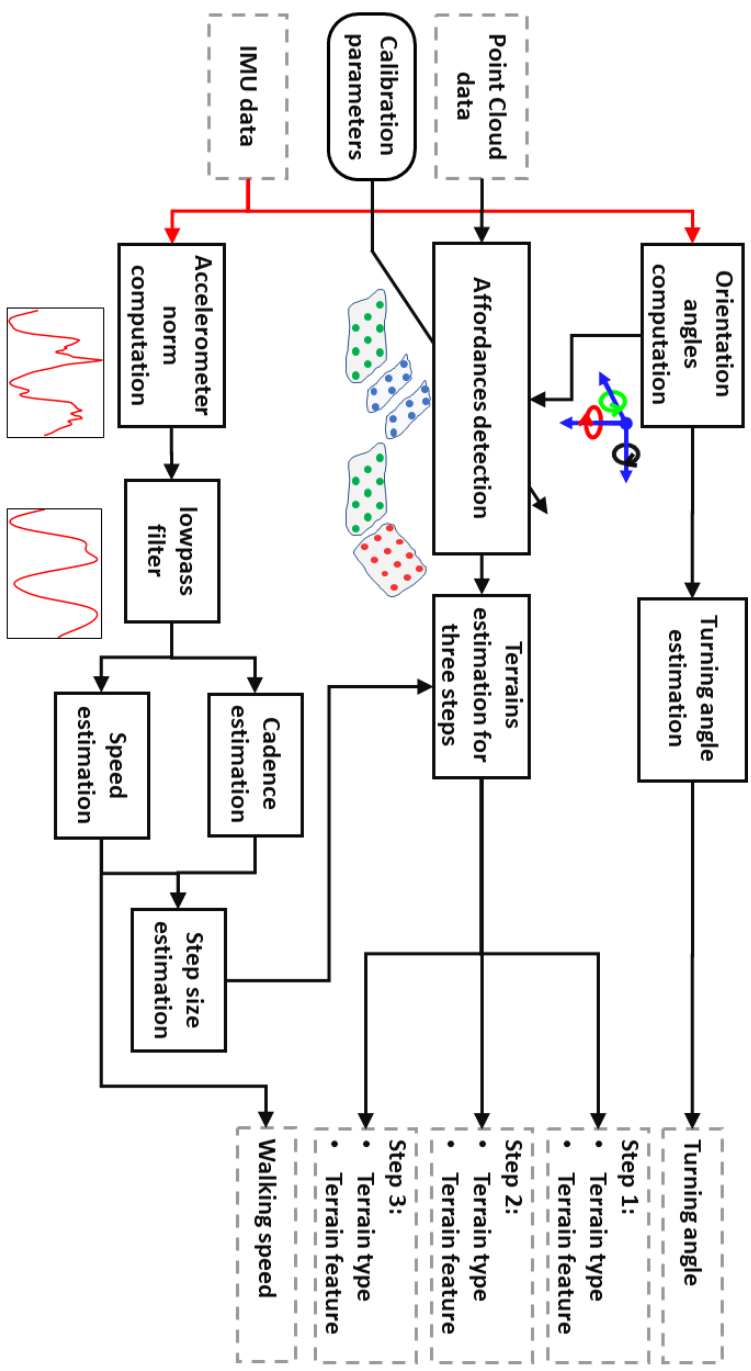


Figure 6.1: A comprehensive block diagram that shows terrain detection algorithm linked to the step size estimation approach in order to achieve terrain detection for several steps with adaptive step sizes. The inputs that are point cloud IMU data which are both obtained from the Realsense D435i depth camera, while the calibration parameters are only set once at the beginning of the trials. The outputs consist of terrain detection for three steps ahead with adaptive to the user's step size. Also, the walking speed and the turning angle are set as outputs. In between, the inputs data passes through multiple processes to achieve the targeted outputs.

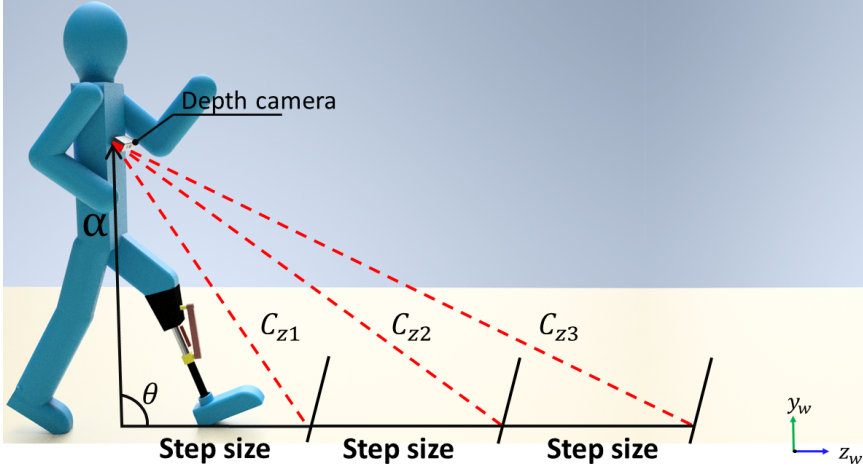


Figure 6.2: The predicted regions for each upcoming step sizes based on the determined thresholds C_{z1} , C_{z2} and C_{z3}

method introduced in 4.2.2.a computes three thresholds (C_{z1} , C_{z2} and C_{z3}) shown in Fig. 6.2 and using triangulation equations as following:

$$C_{z1} = \sqrt{\alpha^2 + S_t^2 + 2 \alpha S_t \cos \theta} \quad (6.1)$$

$$C_{z2} = \sqrt{\alpha^2 + 2S_t^2 + 4 \alpha S_t \cos \theta} \quad (6.2)$$

$$C_{z3} = \sqrt{\alpha^2 + 3S_t^2 + 6 \alpha S_t \cos \theta} \quad (6.3)$$

$$\theta = 90^\circ - S_w \quad (6.4)$$

where α is the camera height, S_t is the step size and S_w is the geometrical feature of the detected affordance' that represents the plane orientation. After computing the thresholds, the points cloud that has z_c components lower than C_{z1} are considered in the first step, and the point cloud that has z_c components lies between C_{z1} and C_{z2} are considered in the second step and so on with the third step. Finally, in order to avoid outliers, the estimated step size is thresholded between 1 and 0.3 m.

6.3 Preliminary results

Typical system ROI for three steps with two different step sizes manually assigned (0.5 and 1 meter) is displayed in Fig. 6.3. The points are projected over the indicated area with yellow tape markers which is 1 m wide and 1.5 m long. The marked area is divided horizontally into three regions with 0.5 m long each. The point cloud that belongs to each step is given a specific color for visualization purposes. Black, orange, and cyan are the assigned colors for the points that lie in the first, second, and third steps respectively. It is obvious from the figure that with a step size equals to 0.5 m the points belonging to each step lie within each region. However, with a 1 m

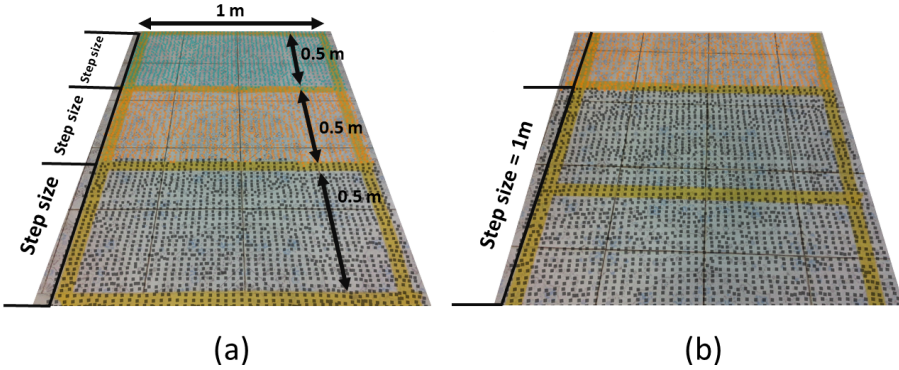


Figure 6.3: Typical system ROI output for three steps during stand still. The points cloud is projected over a marked area with yellow tapes. Black, orange and cyan represent the points cloud colors assigned for the first, second, and third steps respectively. Two different step sizes are manually set to be tested over the same marked area: (a) 0.5 m and (b) 1 m.

step size, the points that belong to the first step (black points) took two consecutive regions which cover 1 m (see Fig. 6.3).

Fig. 6.4 shows the possible output information obtained from our system captured one step away before the transition from G to DR locomotion modes. At that moment, the step size equals $(1.06/1.7=0.63\text{ m})$ and it is fed to the terrain estimation for three steps process to allocate a region for each step, determine the points cloud that belongs to each region, and compute the most probable terrain at each step. Then, it links the geometrical features of the detected affordances to their corresponding steps region. The final terrain information for the three steps at that moment is displayed at the bottom left window of the figure.

By combining the speed, turning angle, and terrain information, we could track the human activity during locomotion over different kinds of terrain as shown in the top left window of Fig. 6.4. More interesting plots are obtained with the satellite images for the two outdoor paths used in Chapter 4 experiments. The plots are displayed in Fig. 6.5 and 6.6. The different colors of the plotted path reflect the detected terrain type over that portion of the path. The colors are assigned corresponding to the color labels given in Table 3.1. Such locomotion activity tracking can be useful for clinical trials or testing assistive devices (prostheses and exoskeletons) over different terrains.

6.4 Prediction time

In order to close the loop with the information presented in Chapter 2, we need to compute the prediction time of our system, and it has been defined as the difference between the critical time t_c , i.e. the time when the locomotion mode changes or should change due to terrain variation, and the time when this change is actually detected, i.e. the detection time t_d . Therefore, it can be computed according to the

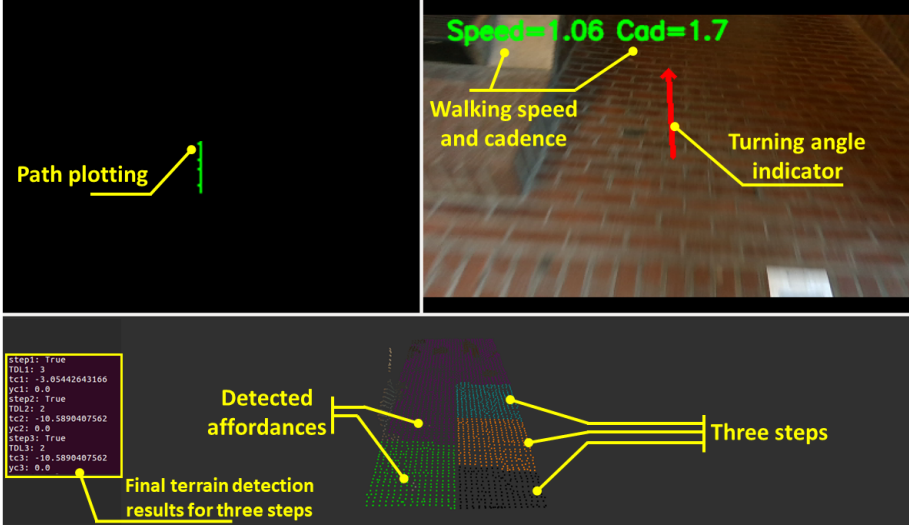


Figure 6.4: Different system outputs that can be obtained from combining the algorithms developed in Chapters 4 and 5. This allows to visually inspect different outputs of the system. The top windows show the path plotting, turning angle, speed and cadence. The bottom window visualizes the points cloud with the detected affordance and the region allocated for each step. Also, the final terrain detection results are obtained for the upcoming three steps with geometrical feature of each terrain such as terrain slope and height labeled with t_c and y_c respectively.

following equation:

$$T_{pred} = \frac{D_{range}}{W_s} - \frac{1}{f_{rate}} \quad (6.5)$$

where D_{range} is the maximum detection range in meters, W_s is the walking speed in m/s, and f_{rate} is the frame publishing rate per second. Therefore, if we assumed that the $W_s = 4$ Km/h (1.11 m/s) which is the same value assumed in Chapter 2, and the maximum detection range (D_{range}) of the used depth camera equals about 2 m, while the frame rate of our algorithm (f_{rate}) equals 11 fps on average. The prediction time of our system will be = 1.7s.

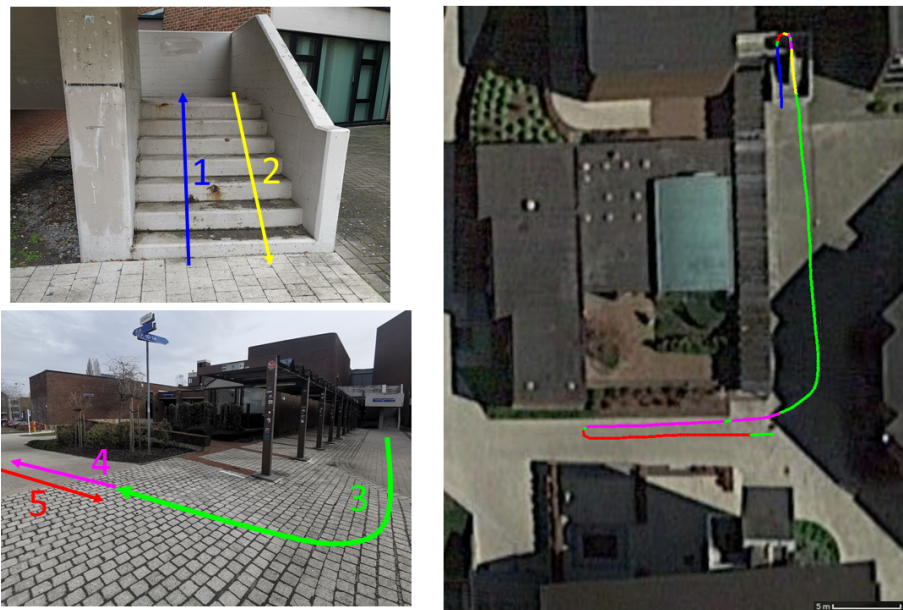


Figure 6.5: Path plotting of a conducted trial in Chapter 4 of the first outdoor path over satellite image of the same path. The path is plotted with various colors which refer to the terrain type observed over that portion of the path. The colors are compatible with the color labeling introduced in Table 3.1.

6.5 Conclusion

In this chapter, we introduced the terrain detection algorithm for multiple steps with the user's adaptive step sizes. The instantaneous step size was computed using the speed and cadence estimation algorithms presented in the previous chapter. Besides the terrain information for the three upcoming steps, we set the speed and turning angle as outputs. Such information was used to track the user's activity over several terrains. This could be useful for evaluating assistive devices' performance over several terrains or used for indoor localization applications. Finally, the prediction time of the system is determined.

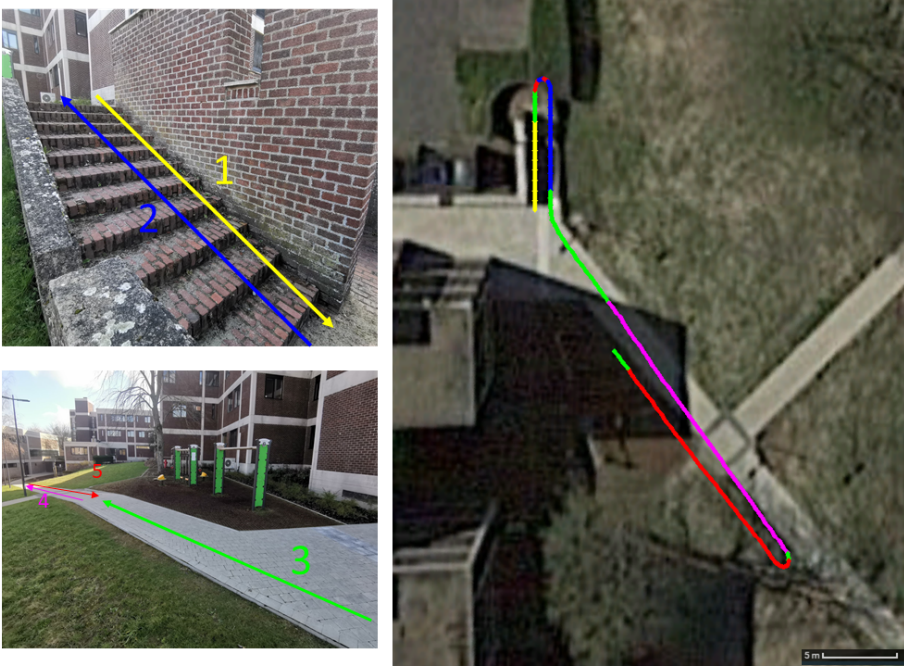


Figure 6.6: Path plotting of a conducted trial in Chapter 4 of the second outdoor path over satellite image of the same path. The path is plotted with various colors which refer to the terrain type observed over that portion of the path. The colors are compatible with the color labeling introduced in Table 3.1.

General discussion and conclusion

7

This chapter steps back and discusses the work carried out during this thesis. In particular, we explore and discuss the contributions of this thesis. We also review back the research question and present our answers based on the achieved outcomes. Finally, we conclude this work and present our perspectives for future development.

7.1 Contributions

Healthy leg joints produce different kinetic and kinematic patterns for various encountered terrains in everyday life such as stairs, ramps, and ground. As consequence, the active assistive devices (prosthesis and exoskeleton) are expected to generate such healthy patterns for each terrain. However, switching from one locomotion mode to another, requires a timely transition, sometimes even before the critical time or event. Therefore, some of these devices are equipped with proprioceptive sensors for predicting the transition between different locomotion modes. However, with such sensors, the transition is detected after the critical event or a few milliseconds before. As consequence, the occurrence of falls and injuries becomes more ubiquitous due to the delay. Interestingly, vision sensors such as depth camera can provide more predictive information about the surrounding environment. However, this information needs to be processed and exploited to determine the upcoming terrains for several steps. This will provide an adequate amount of time to the device controller for reacting and preparing the locomotion plan for some steps which could emulate more healthy gait pattern. Moreover, estimating the geometrical features of the detected terrain is essential for achieving adaptive assistance. Because the terrains in the urban environment are exposed to the occlusion by other walkers, the implemented system also should demonstrate its ability to recognize the upcoming terrains and their geometrical features even with partial occlusion conditions. Therefore, by considering such objectives, the following list enumerates the main contributions of this work:

- A computer vision based algorithm is implemented which processes the point cloud data obtained from a wearable stereo camera to achieve terrain identification. The sensor and the algorithm are carefully selected and built based on

the reviewing results of more than one hundred articles that use various types of proprioceptive and/or exteroceptive sensors for locomotion mode and terrain identification.

- The algorithm demonstrates the ability to identify several types of terrain for multiple steps in front of the user. Indeed, 98% and 92% are the grand average accuracies that are recorded for the first step and second steps respectively, when the system was tested with multiple indoor and outdoor environments with clear path conditions. Furthermore, the system is equipped with a step size estimation approach based on walking speed and cadence to determine the upcoming terrain for some steps adaptive to the user's stride length during ground and ramp locomotion modes.
- The algorithm computes the geometrical features of the detected terrain of each upcoming step. The results show that the estimated features come with a low error rate for most of the terrain types when compared to the actual values.
- Moreover, our system shows robustness in estimating the terrain type and geometrical features within the partially occluded environment, at least for a single upcoming step. This was validated through several pilot experiments that were conducted in partially occlusion conditions with multiple indoor and outdoor paths.
- Additionally, the algorithm provides the instantaneous walking speed and turning angle estimation using a single IMU sensor attached to the user's chest. Accordingly, our speed estimation approach can be useful to evaluate the assistive device performance through testing it in various environments without the need for a special set up or treadmill. In contrast, the proposed method for estimating turning angles might be used to adjust the stiffness of an active locomotion assistive device at the turning moment.

7.2 Discussion

This section reflects on the system's performance with respect to the state of the art. Additionally, it discusses some other aspects regarding the conducted experiments and data sharing. This section, also, expands on the system's strength and limitations while offering some possible solutions.

System performance w.r.t. the state of the art

Many terrain detection systems that use depth vision have recently emerged for locomotion assistance. These systems apply various algorithms in order to accomplish terrain identification in front of the user. However, we observed several limitations of such systems. For instance, they are targeting terrain identification in the whole region of interest without predicting the terrain information for specific upcoming step(s) which could be adaptive to the users' stride length. Moreover, the system detection capability of a partially occluded terrain is not addressed by these systems

which is an essential factor in urban environments. Neglecting such details can cause an improper reaction of an active assistive device in several instances of the encountered terrains. Therefore, to the best of our knowledge, the presented system in this thesis can be considered as the first depth camera-based system that provides terrain identification and geometrical feature estimation for several steps with robustness to partial occlusion. Accordingly, to position our results with respect to the state of the art, we could only compare the first step accuracy detection results in clear path conditions with the most recently introduced system for active locomotion assistive devices. In that case, we observe that our approach reaches competitive accuracy results with a grand average of about 98% which is similar to that recently published approach. Moreover, our results might be considered as being more robust since we tested the system with two indoor and outdoor paths while the recent approaches presented their results based on single indoor and outdoor paths.

In contrast, the speed and step size estimation methods came with low error rates that were equivalent to those found in prior studies which used a single IMU with a similar sensor configuration. Moreover, our methodology has a higher publishing rate (2Hz) due to the relatively small window size (1 second) applied to the input signal for features extraction while the previously introduced methodologies use window size that can reach up to 10 seconds. Such a high publishing rate hold promises for combining this approach with the terrain detection algorithm in order to offer a terrain estimation that corresponds to the predicted foot placement for the upcoming steps which is adaptive to the user's immediate step size adjustments.

Another important parameter is the prediction time. For the same walking speed, the system prediction time has been estimated to be about 1.7s. This is within the same range as the equivalent exteroceptive systems that are listed in Table 2.2. This time can be improved using a longer-range depth camera. However, this estimated prediction time should be sufficient for the assistive device to switch to the next upcoming task in a timely manner.

The introduced terrain detection algorithm can, also, be used for visually impaired people's assistance. Indeed, several approaches are presented in the literature that depends on the depth and/or normal camera for terrain identification in front of the user [140–144]. However, the latter may not be the best option for detecting geometrical terrains since textures can substantially impact them. For instance, crosswalks on a road can be detected as staircases by the algorithms that rely on the standard camera while the depth vision-based approaches can overcome such challenges [141]. Therefore, most of the introduced methods use depth cameras for staircase identification. Interestingly, our algorithm offers stairs and ramps recognition as well as the possibility of determining the next terrain with respect to the user's stride length.

Experiments and data sharing

For the terrain recognition system assessment, experiments have been conducted on four paths (two indoor and two outdoor). Each path has geometrical features that are different from the others, see Table. 4.1. Regarding the experimental trials, each participant went through the four paths two times. This guarantees the occurrence of a natural way of walking while wearing the camera due to the many trials per

participant, and adds more confidence to the obtained results for not being biased toward a non-natural way of walking. Furthermore, data were collected over several days and under different weather conditions. Consequently, this brought more challenges to the algorithm to prove its robustness to such ecological variations.

For assessing the speed, cadence, step size, and turning angle estimation methods, the experiments were conducted in lab-oriented environments in order to have an accurate ground truth computation of the collected data. Also, since we are relying on an IMU for estimating these parameters, environmental changes should have no influence on such a sensor, in contrast to the depth camera which is directly impacted by lighting conditions.

Unfortunately, we were unable to find an online dataset collected by other researchers to evaluate our algorithm with independent data. There is indeed a lack of online datasets in the community working with human locomotion assistive devices. Accordingly, to enable other researchers to benefit from our collected data for evaluating their methods or developing new approaches, the raw collected data have been shared on an open online access dataset (see Appendix C). The data are packaged within ROS environment that provides a handy tool to record and re-stream several sources of data simultaneously as a single file.

System strengths and limitations

The terrain detection algorithm framework that relies on recognizing the terrain based on manually extracted features is one of the main strengths of the presented approach, since it can bring some guarantees about the system adaptability to new environments. This contrasts with black box-based techniques such as a deep neural network that relies on the raw data as input. These methods may fail in new or untrained environments. Moreover, the algorithm shows noticeable robustness to various lightning conditions in outdoor environments during sunny and cloudy lighting conditions; and low illumination in the case of the indoor environment.

In contrast, we also observed some limitations regarding the system performance. For instance, during the user turning, the detection algorithm did not work as expected. This can be noticed in the reconstructed path (Figures 6.5 and 6.6) that show several wrong predictions when the user was turning on flat ground nearby the stairs. However, the control engineer can benefit from the turning angle estimation to eliminate these inaccurate predictions by switching to the Ground mode while the user is turning. Other minor failure cases can be handled by relying on the geometrical features estimation of the detected terrains or using some advanced filtering technique through incorporating history information about the prior detected terrains [76].

For the speed, cadence, step size, and turning angle estimation, a typical advantage is that they are independent of the environmental conditions. They are indeed proprioception-based methods measuring human kinematics without the need to interact with the surrounding environment. However, these methods do also suffer from some drawbacks. For instance, any backlash in the sensor belt can cause extra vibrations that add noise to the acceleration readings. This leads to inaccurate prediction in the speed and cadence which has a detrimental impact on the step size estimation. Therefore, an alternative configuration would be to rely on

an IMU sensor that is directly built-in in the assistive device to estimate the walking speed, cadence, and step size. However, this approach brings its own challenges, since this is likely that the corresponding segment will be less steady than the user's chest. For the turning angle, a major drawback is the intrinsic delay in the turning detection. In fact, it detects the change in the angle after the turning event occurred. This implies a negative prediction time. A promising methodology is presented by [199] that integrates several sensory information using a machine learning-based approach in order to predict the turning event with a positive prediction time.

7.3 Research questions and (partial) answers

At the beginning of this thesis, we detailed three main questions to be addressed in the context of this thesis (see Section 1.2). Therefore, we review them based on our observations from the experiments conducted in this thesis.

- *How can we achieve terrain detection for several upcoming steps?*

Obtaining terrain information for some steps in advance allows the prosthesis or exoskeleton to present early reactions and avoid the occurrence of falls of disabled people. This requires sensory capabilities that have direct interaction with the surroundings like exteroceptive sensors. In particular, a depth camera can provide rich information about the environment for more than one meter of distance without being heavily influenced by the lighting conditions or textures. Processing the obtained point cloud data from such sensors can lead to estimating the environmental features such as height and orientation, and with the proper classification method, the locomotion affordances can be predicted. However, to estimate the locomotion affordances for some steps, we present two processing methods based on the detected terrain type. For the staircase, the affordances are already physically separated. Therefore, each detected step of a stair could correspond to an upcoming step that will be taken by the user. However, for physically connected affordances such as the ground-ramp case, the proposed cloud splitting, and merging methodology provides a promising separation between ground and ramp affordances. This is followed by segregating the ROI into equally sized regions corresponding to the user's step size which could be assumed to be fixed. Otherwise, it could be estimated based on his/her walking speed and cadence that are computed, for instance, by exploiting the IMU data of the wearable camera as proposed in this thesis. Each determined region could represent a single step; therefore, the terrain type for each step can be determined depending on the labeled cloud contained in that region.

- *How can we achieve geometrical features estimation for the detected terrain in the upcoming steps?*

Observing the geometrical features of the detected terrain allows the assistive device to tune the amount of power injected to achieve smoother locomotion over various terrain types. Accordingly, the points cloud data obtained from the depth camera construct a 3D shape of the environment in front of the user.

Therefore, normal estimation, cloud segmentation, and plane fitting are typical processes that could be applied to extract environmental features of the affordance(s) ahead of the user such as orientation and the terrain centroid coordinates with respect to the user's feet. While these features can be used to obtain the terrain type (stair, ramp, or ground), we could rely on them to determine the geometrical feature of the locomotion affordances for some steps in advance.

- *How and to what extent can we obtain the terrain type identification and geometrical features estimation even with partially occluded environment?*

Facing partially occluded terrains by some people in the urban context is very probable to occur. Therefore, the implemented terrain detection algorithm should demonstrate robustness to such a setting so that the assistive device could keep delivering the expected support to disabled people. According to the assessment method presented in this thesis, we observed that depending on 3D construction of the detected affordances allows the terrain to be seen by the camera even with partial occlusion which could successfully lead to feature extraction and terrain identification.

7.4 Conclusion and perspectives

In this thesis, we present a terrain detection system that predicts the locomotion affordances in front of the user. It applies several processing steps on the point cloud data obtained from a depth camera to extract the environmental features and accomplish terrain identification. The locomotion affordances are further processed to obtain the terrain type for more than one step. Based on the IMU data, a step size estimation algorithm is presented so that the user's predicted stride length can be fed to the terrain estimation approach for several steps to accomplish terrain prediction for the next step with respect to the user's step size. Moreover, the geometrical features of the detected affordances such as slope or height are also estimated and linked to each upcoming step info. Furthermore, the algorithm offers the user's walking speed and turning angle estimation.

The system was evaluated against several indoor and outdoor paths with rich terrains that have different geometrical specifications. Furthermore, the experiments were conducted with clear and partially occluded path conditions and several participants. The system achieves more than 95% of terrain identification accuracy for the first step for all locomotion modes with grand average accuracies of 98% and 92% for the first and second steps respectively. In contrast, small error rates were observed for the other estimated parameters such as geometrical terrain, walking speed, and step size. The achieved results are competitive with recently published papers that use the same sensor category and/or configuration. Fig. 7.1 display a block diagram that shows how the carried out work during this Ph.D. thesis was distributed over the chapters.

While this thesis addresses the geometrical terrain identification multiple steps in advance for locomotion assistive devices, recent studies show that some non-geometrical terrains such as small rocks and gravel impact the locomotion patterns of healthy and disabled people. Accordingly, research is currently ongoing by a Mas-

ter student under our supervision to identify and detect the non-geometrical terrains that mostly influence human locomotion patterns. In this case, we are relying on a standard camera to capture different images from the outdoor environment of non-geometrical terrain and apply a terrain identification algorithm based on Convolutional Neural Network (CNN).

Future work concerns further development of the system software and the hardware toward designing a wearable terrain detection system that accomplishes real-time terrain identification with an acceptable refresh rate. On the software side, relying on multiple regions of interest with small widths can reduce the point cloud size and save more computational time. On the hardware side, the lightweight and compact size of the Realsense D435i stereo camera makes it a good option to be used in wearable assistive devices. It provides rich information about the environment in the point cloud and standard image formats which makes it beneficial for geometrical and non-geometrical terrain identification. Based on the experiments conducted in this thesis, we found that it has a good ability to work in indoor and outdoor environments without being heavily impacted by the change in the illumination condition. For the mini-computer, Jetson nx from NVIDIA could be considered an interesting option. Indeed, it can be used for high-performance AI systems like commercial robots and medical instruments. Consequentially, such a wearable device consisting of a small camera and mini-computer with some accessories could have a potential interest to be used by several categories of disabled people like amputees with active locomotion assistive devices or visually impaired people. Finally, it is worth mentioning that with the rapid advancement in technology, it is likely that in the near future, the presented algorithm might possibly run on the upcoming smartphone generation.

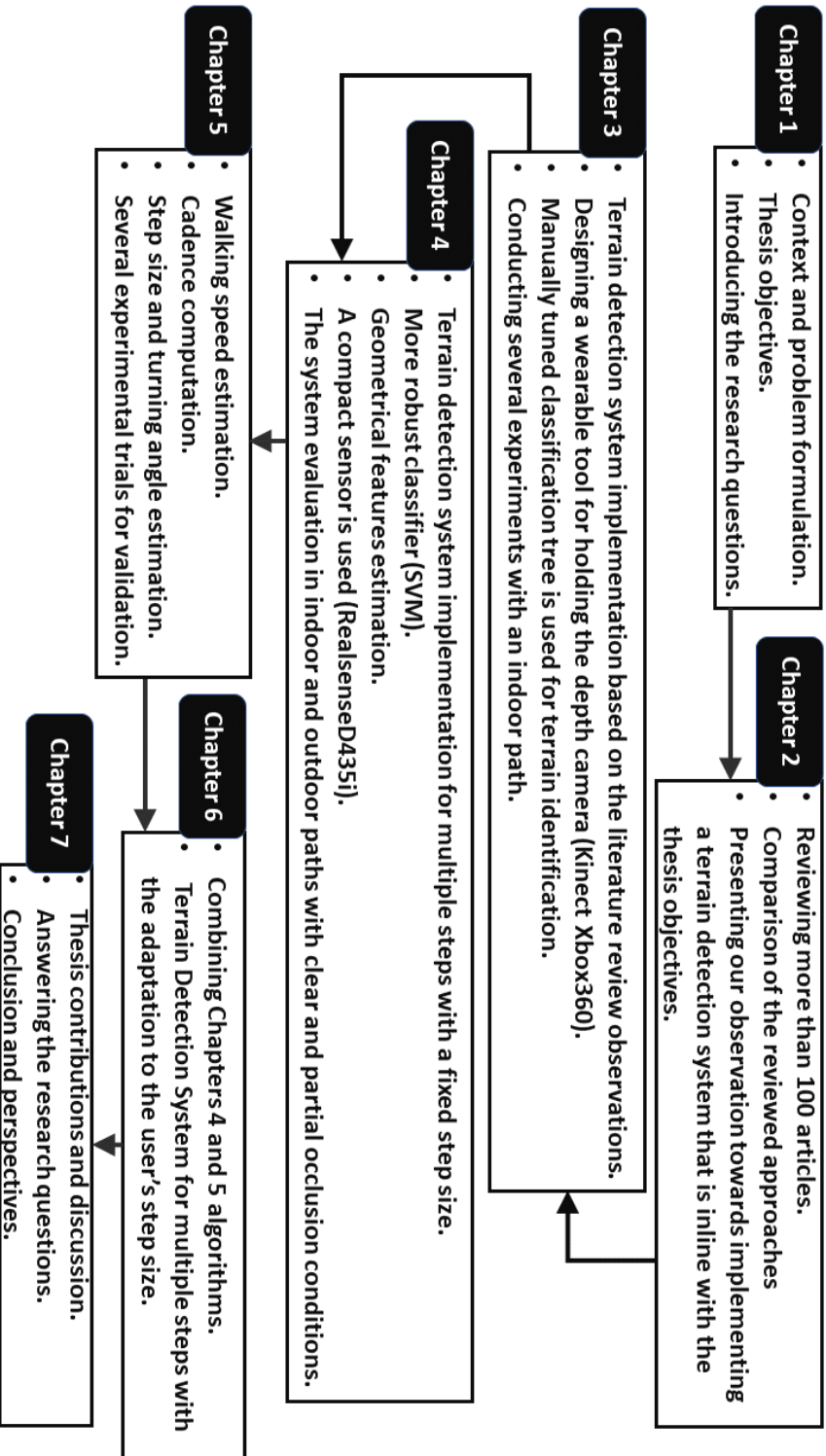
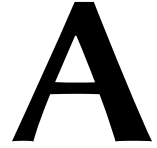


Figure 7.1: Thesis structure showing the organization of the chapters and their respective dependencies.

Appendix

Experimental data of terrain detection system for several steps with classification tree



This chapter presents experiments and results conducted with terrain detection approach for several steps using classification tree instead of the SVM classifier. This version of the algorithm is not included in the main text. It applies the same processes explained in Chapter 4 with some differences. For instance, In order to make the system robust to the intrinsic sensor noise of the IMU, θ_c was kept constant and equal to a value calibrated when the subject was standing still, if the detected locomotion mode was ground or up-/down-ramps. In both stairs modes, θ_c was extracted in real-time from IMU data. The number of detected planes was triggering the system to switch between fixed and instantaneous IMU data. In all cases, α was calibrated by measuring the distance between the camera and the ground once the device was mounted on a given subject. This approach has some drawbacks because it will be influenced by large changes of the camera orientation during locomotion can lead to wrong detection in ramp and ground modes. In additional to the plane orientation angle S_w and the plane height with respect to the user's feet H_w , an extra features was added If two different adjacent planes are detected, their relative distance in the forward direction is computed as $D_z = |o_{z1}^w - o_{z2}^w|$. This extra feature will turn out to be useful for improving the detection accuracy of staircases by preventing the system from getting tables and low leveled shelves on the ground mixed up with stair steps. However, in the algorithm implementation introduced in Chapter 4; we computed the geometrical features of the detected terrain instead which allows more flexibility in dealing with affordances with various heights. The three features passed to the classification tree are pictured in Fig. A.1. The tree applies several tests on these features which leads to various terrain predictions as shown in Fig .A.2. Similarly, a specific color is assigned to each detected type of terrains according to the color labeling introduced in Table 3.1.

In the following sections, we explain the experimental methods and discuss the achieved results with this algorithm.

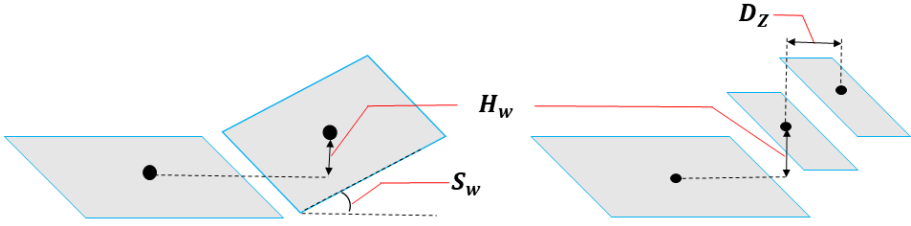


Figure A.1: Features fed to the classification tree to achieve various terrain recognition. S_w is the plane orientation angle, H_w is the plane height with respect to the user's feet and D_z is the relative distance between two consecutive planes in the forward direction

A.1 Experimental methods

An indoor path on the UCLouvain campus of Louvain-la-Neuve was selected to assess the system performance (see Fig. 3.6). The path contains different kind of targeted terrains such as ground, stairs, and ramps. The wearable tool shown in Fig. 3.5 with Kinect camera is used in this system implementation. The system was tested in the two following conditions:

- Uncluttered environment: the system was tested with six healthy subjects (4 males and 2 females, between the ages of 27 and 32). Each individual wore the device and walked through the specified path while it was free of other people, at their preferred walking speed.
- Cluttered environment: in order to reproduce a real life scenario in which the walking path is not always free of other people, we further conducted pilot tests to assess the detection robustness with partially occluded footages. This is important since real urban environments could be cluttered with other people or obstacles. Accordingly, two healthy subjects participated to this pilot experiment. The first subject played the user role and was thus wearing the device, and the other subject played the role of the impediment and was thus walking in front of the user on the same path. This experimental condition was conducted twice. In the first trial, the leader walked very close upfront of the user (approximately 0.8 to 1.2 m). In the second trial, the leader walked a bit further away from the user (approximately 1.2 to 2 m).

A.2 Accuracy estimation

In order to estimate the system accuracy in detecting each locomotion mode in uncluttered environment, the recorded data with the 6 subjects were categorized based on the five locomotion tasks identified above and labeled as ground (G), down-ramp (DR), up-ramp (UR), ascending stairs (AS) and descending stairs (DS). Ground truth labeling was performed by an expert (first author of this paper), using the RGB footages recorded during the experiments. A typical evolution of T_{actual} vs. $T_{1,detect}$ for the full path (including all locomotion tasks) is pictured in Fig.A.3. Computing

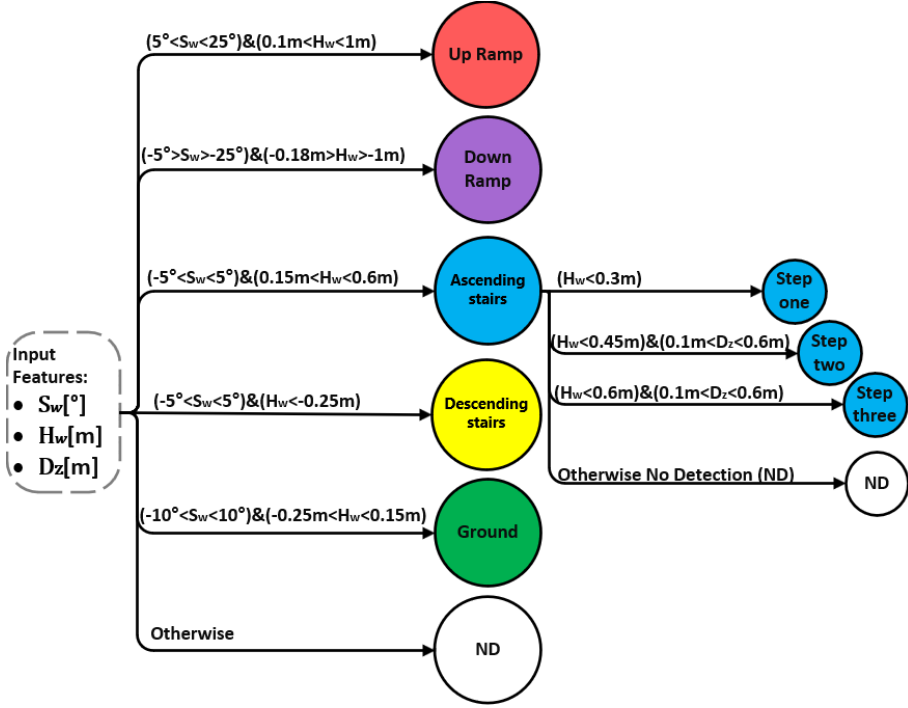


Figure A.2: The classification tree examines the extracted features S_w , H_w , and optionally D_z , against several conditions to achieve terrain identification. First, it begins by testing the features for the ramp conditions, then for the stair conditions, and finally for the ground conditions. The plane is considered to be an obstacle if its features lie outside all the above constraints. Colors given for each type of detected terrain correspond to the colors assigned to the point cloud of raw data for visualization purposed (see e.g. Fig. A.4). If ascending stairs are detected, a further step is applied using the third feature D_z to confirm the validity of the steps being successively detected. For instance, the second step is considered as being an obstacle (No detection) unless the first one is properly detected and similarly for the third step (see the text for more details).

the percentage of samples where $T_{1,detect}$ and T_{actual} are equal over the experimental trial gives an overall estimate of the system accuracy for each of the three steps ahead of the user, independently. Similarly, confusion matrices between each of the conditions can be computed by reporting the fraction of samples where both vectors disagree.

A.3 Results and discussion

A.3.1 Uncluttered environment

Typical detection samples for different kind of terrains are provided in Fig.A.4 , with the color code corresponding to the one of Fig.A.2. It can be seen that the system pro-

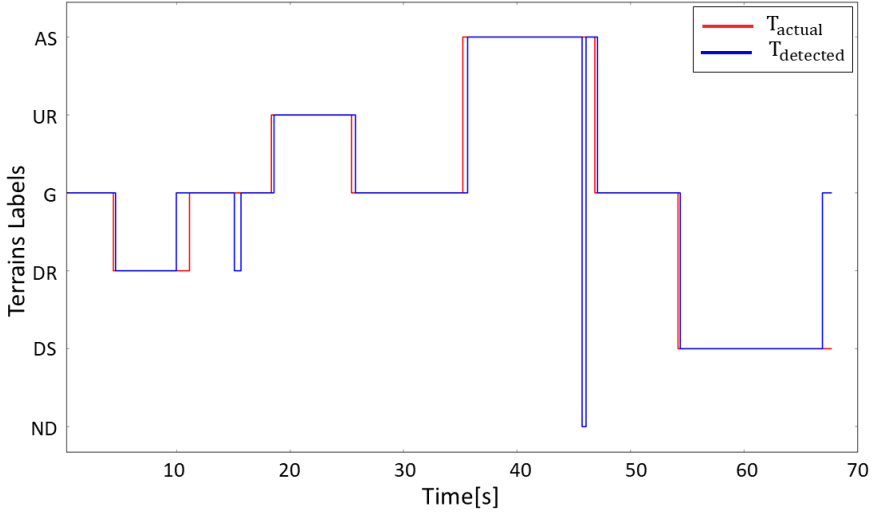


Figure A.3: Actual (T_{actual}) vs. detected (T_{detect}) terrains recording in a typical trial involving ground (G), down-ramp(DR), up-ramp(UR), ascending stairs(DS) and descending stairs(DS).

vided a good separation of ground planes at the transition between different kinds of terrains. In particular, the separation between the ground and ramp was well detected although there was no physical separation between them. Such accurate separation holds promise if fed to the controller of an active assistive device for locomotion (such as an exoskeleton or a prosthesis) since the new locomotion mode should be triggered at the right time. The average system accuracy was computed for each locomotion mode and reported in Fig.A.5 for one to three steps ahead. To have a better insight about the wrong detections, we computed the confusion matrix for the three steps, and they are provided in Tables A.1 to A.3. For the first two steps, the system achieved above 90% of detection accuracy for all kind of terrains. However, the system reached lower accuracy for the third step, corresponding to the one being the furthest away from the user. In particular, the detection accuracy got degraded for detecting ascending and descending staircases. This was mainly due to the fact that in this case, the sensor was lacking a complete vision of the whole staircase, since the edge of the third step was not detected most of the time. The ND (no detection) label corresponded indeed to the most frequent confusing detection in these cases. A possible solution would be to tilt the sensor up a bit more in order to have a full vision of the third step, but this would deteriorate the capacity to detect the first step since the terrain just in front of the user would be out of the camera field of view.

A.3.2 Cluttered environment

To the best of our knowledge, testing vision-based locomotion detection on terrains partially occluded by other people has never been addressed so far. However, it is a matter of fact that typical urban environments are crowded most of the time, and

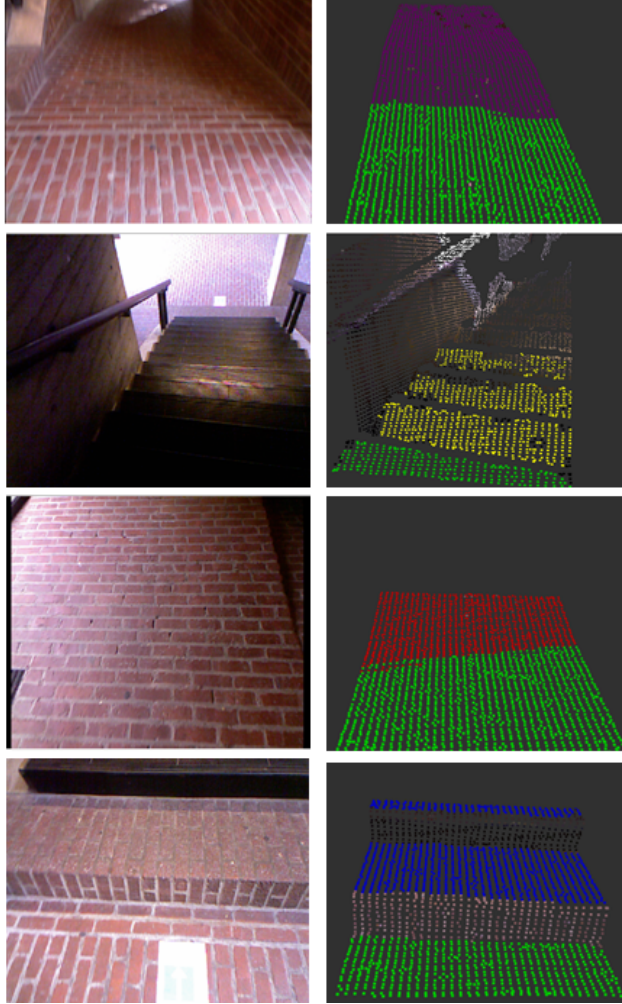


Figure A.4: Detection samples of different terrains. The left column shows real RGB footages while the right column shows the algorithm recognition results. Green points refers to ground detection, while red and purple corresponding to up-ramp and down-ramp respectively. Steps of ascending and descending stairs are pictured in blue and yellow points respectively.

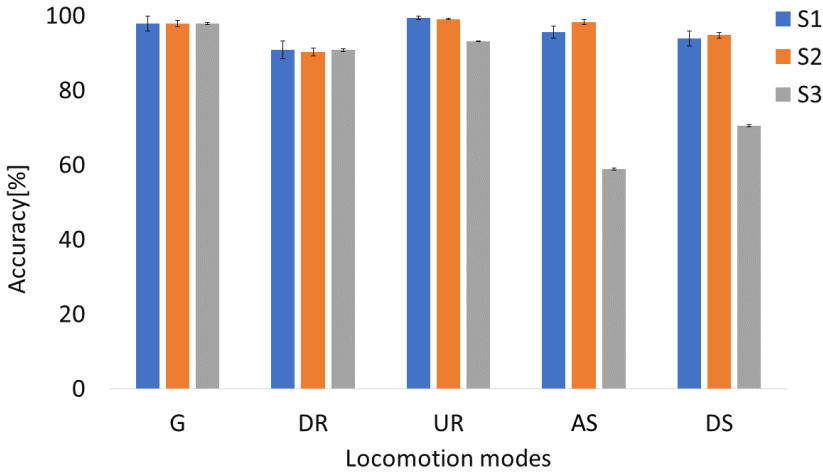


Figure A.5: Average accuracy for each locomotion task and for three steps in front of the user in uncluttered environment. S1, S2 and S3 refer to steps 1 to 3. Whiskers represent the standard error of the mean, obtained across the 6 different users. Labels correspond to G: flat ground; DR: down-ramp; UR: up-ramp; DS: descending stairs; and AS: ascending stairs.

that people tend to like walking in a group. Accordingly, we assessed preliminary performance of our system while the path was partially occluded by another person, walking at two different distances upfront. Detection estimates corresponding to various locomotion modes are displayed in Fig.A.6. The accuracy was calculated for both trials, and again for three steps upfront of the user, as illustrated in Fig. A.7 and A.8. When the leader was close to the user (0.8-1.2m distance between them), the system achieved lower accuracy in comparison to the results reported for a clear path, mainly for the third step upfront. This is likely because the leader occluded any relevant visual information for this third step. In the second trial, where the leader walked further away from the user (1.2-2m distance between them), we obtained similar results for the first two steps as with clear path, and again a slightly lower accuracy for the third step.

Table A.1: Confusion matrix for step one. (ND) refers to No Detection.

	G	DR	UR	AS	DS	ND
G	98.14	0	1.85	0	0	0
DR	9.54	90.45	0	0	0	0
UR	0	0	99.5	0.5	0	0
AS	2.62	0	0.32	95.08	0	1.96
DS	6	0.8	0	0	93.14	0

Table A.2: Confusion matrix for step two

	G	DR	UR	AS	DS	ND
G	98.14	0	1.85	0	0	0
DR	8.04	90.45	0	0	1.5	0
UR	0.52	0	98.95	0.52	0	0
AS	1.02	0	0.68	98.29	0	0
DS	4.45	1.61	0	0	93.92	0

Table A.3: Confusion matrix for step three

	G	DR	UR	AS	DS	ND
G	98.14	0	1.85	0	0	0
DR	9.14	90.85	0	0	0	0
UR	8.612	0	91.38	0	0	0
AS	0.36	0	4.39	59.34	0	35.89
DS	1.62	6.91	0	0	69.91	21.54

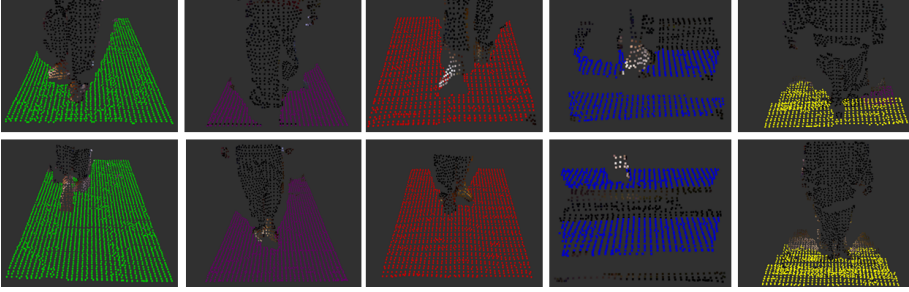


Figure A.6: Detection results during the five locomotion tasks while the terrains were partially occluded by another subject. The first row of images show the detection results while the leader subject was 0.8-1.2m upfront of the user (first trial). The second row shows detection results while the leader was 1.2-2m upfront of the user (trial 2). The color code is the same as for Fig.A.4.

A.3.3 Discussion

The system we implemented showed remarkable detection results in both uncluttered and cluttered environments. Our algorithm is built upon a framework that has already been tested by others [35], although it requires a priori less computational resources because we transformed only the detected plane features into the local world reference frame, rather than the whole point cloud. Consequently, this approach is promising for wearable systems with limited processing capabilities. The other major contribution of our study is to outline a framework that can provide a probabilistic estimate of the terrain to be traversed up to three steps ahead of the user. This predictive capacity is likely to be critical if our system is coupled to a locomotion assistive device, in order to trigger the transition for one locomotion mode to another in a smooth and timely fashion. A possible extension would be to adapt the size of the locomotion stride as a function of the actual one of the user.

However, the current implementation of our system has some limitations. For instance, no real-time camera orientation was used in the flat ground and ramp modes. This means that large changes of the camera orientation during locomotion can lead to wrong detection in these modes. Also, due to hardware limitations of the selected camera, the system cannot work under the direct sunlight, so we could not test our algorithm in outdoor environments. Next, the classification methods that was used uses hand-tuned thresholds that are based on geometric features of typical environments. Although this provides a direct mapping between the environment geometry and the resulting classification, this might lead to poor generalization to other paths with different features.

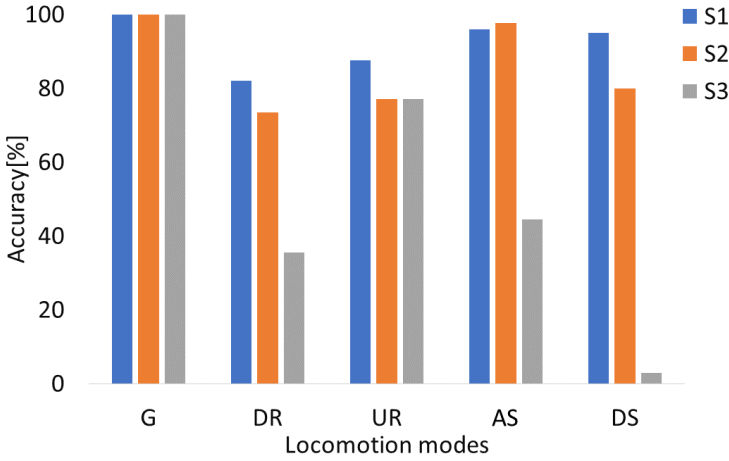


Figure A.7: Accuracy for each locomotion task and for three steps in front of the user in cluttered environment when the leader walked 0.8-1.2m upfront the user (1st trial). S1, S2 and S3 refer to steps 1 to 3. Labels correspond to G: flat ground; DR: down-ramp; UR: up-ramp; DS: descending stairs; and AS: ascending stairs.

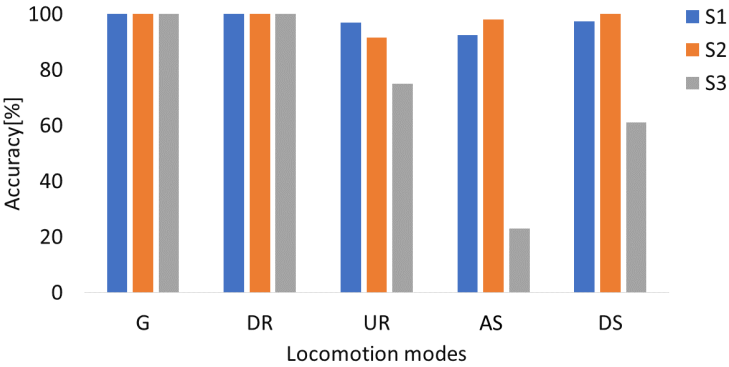


Figure A.8: Accuracy for each locomotion task and for three steps in front of the user in cluttered environment when the leader walked 1.2-2m upfront of the user (2nd trial). S1, S2 and S3 refer to steps 1 to 3. Labels correspond to G: flat ground; DR: down-ramp; UR: up-ramp; DS: descending stairs; and AS: ascending stairs.

Detection accuracy results

B

This chapter presents the detection accuracy results computed for the all participants in the experiments introduced in Chapter 4. The results are recorded for each participants while walking over the two indoor and the two outdoor paths with clear and partially occluded path conditions. The naming abbreviations used for each path are presented in Fig. B.1 while the recorded accuracies are displayed in Fig. B.2.

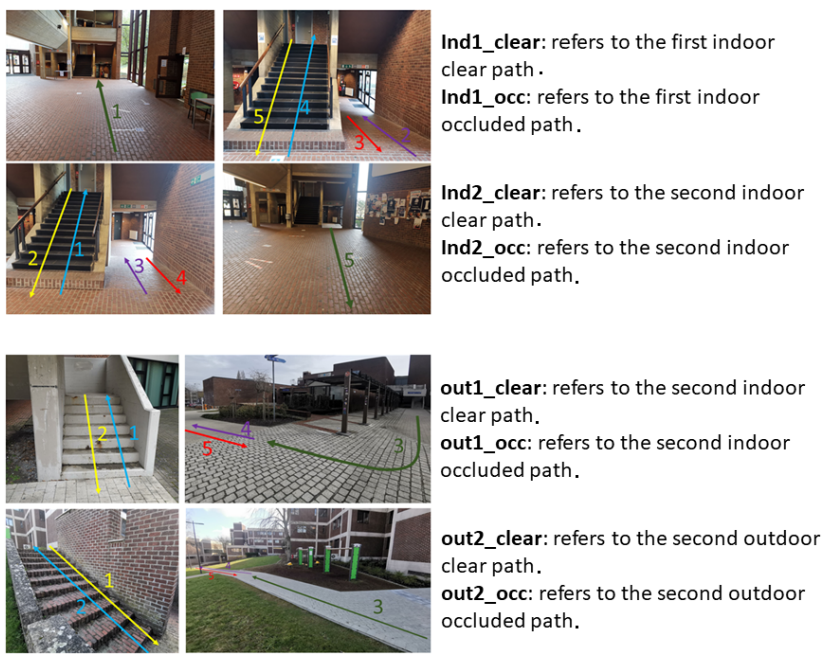


Figure B.1

	Ind1_clear			Ind2_clear			out1_clear			out2_clear			Ind1_oc			Ind2_oc			out1_oc			out2_oc					
	s1	s2	s3	s1	s2	s3	s1	s2	s3	s1	s2	s3	s1	s2	s3	s1	s2	s3	s1	s2	s3	s1	s2	s3			
P1	G	100	100	100	100	100	100	99.54	99.54	99.54	100	100	100	100	100	98.88	98.88	1.12	100	95.6	3.3	100	58.33	83.33	100	100	100
	AS	100	100	8.75	100	93.33	1.33	100	54.84	0	100	91.55	22.54	98.88	98.88	1.12	100	95.6	3.3	100	58.33	20.83	100	90.77	7.69	100	
	UR	100	89.29	82.14	100	100	97.44	97.44	100	91.67	69.44	100	87.23	74.47	93.1	71.26	34.48	100	89.61	64.94	38.71	16.13	12.9	100	87.04	79.63	
	DR	100	100	75.47	97.44	97.44	97.44	100	100	97.46	100	100	78.41	100	100	65.08	87.5	91.67	81.25	100	100	73.34	100	100	88.24	75.65	
P2	G	98.8	98.8	98.8	100	100	100	100	100	100	100	100	100	100	100	98.99	98.99	98.75	100	100	100	100	100	100	95	95	
	AS	96.55	98.85	34.48	96.34	98.78	28.05	100	85	7.5	87.67	95.89	48.32	100	91.91	72.06	99.24	93.18	75	98	84	52	67.54	77.19	58.77		
	UR	91.8	70.49	52.46	100	98.21	92.86	100	78.26	65.22	100	81.16	63.77	43.88	24.49	11.22	55.12	27.56	16.54	76.56	60.94	21.88	94.59	68.47	38.74		
	DR	100	96.77	96.77	100	95	95	100	98.21	95.54	82.67	84.65	81.68	100	98.57	100	100	100	100	100	100	100	95.86	95.86	95.17		
P3	G	78.31	78.31	78.31	100	100	100	95.5	95.5	95.5	91.89	91.89	91.89	95.74	97.87	95.74	98.57	98.57	100	98.39	98.39	96.25	96.25	96.25	96.25	96.25	
	AS	96.77	87.1	18.28	100	98.75	37.5	94.44	66.67	7.41	100	90.67	12	100	100	37.89	99.04	91.35	38.46	100	82	46	100	87.84	6.76		
	UR	100	86.54	84.62	100	90.48	90.48	100	88.46	61.54	100	84.78	80.43	70.65	39.13	25	86.05	33.72	17.44	100	29.09	16.36	100	97.1	85.51		
	DR	100	100	92.73	100	100	100	87.85	86.92	82.24	100	95.6	63.69	56.69	100	98.18	94.55	94.87	94.87	100	100	100	99.48	99.48	99.48		
P4	G	96.59	96.59	96.59	97.39	97.39	97.39	99.44	99.44	98.89	100	100	97.53	94.59	94.59	94.59	99.08	99.08	99.08	99.05	99.05	98.57	100	100	100	100	
	AS	100	88.75	11.25	100	84.29	5.71	100	37.14	0	100	86.71	22.58	100	88.89	11.11	100	83.33	8.33	100	57.89	7.89	100	92.75	5.8		
	UR	100	100	0	100	100	65.71	99.17	99.17	5.79	97.53	97.53	24.69	100	98.51	76.12	100	100	64.71	98.28	98.28	56.9	89.95	89.95	56.61		
	DR	100	100	100	100	91.89	91.89	100	100	100	100	100	100	100	96.23	92.45	100	89.74	76.92	100	100	100	100	100	100	99.46	
P5	G	91.67	91.67	90.48	100	100	100	90.37	90.37	90.37	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
	AS	100	92.05	47.73	100	91.46	67.07	100	89.19	18.92	100	86.67	51.67	100	98.8	56.63	100	94.81	64.94	100	71.43	28.57	98.36	91.8	63.93		
	UR	100	100	100	100	95.24	95.24	100	100	100	100	99.39	99.39	100	100	100	100	100	100	100	100	100	100	100	100	96.17	
	DR	100	96.08	98.04	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	96.17	
P6	G	77.63	77.63	76.32	99.07	99.07	99.07	100	100	99.53	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
	AS	100	99.02	29.07	100	87.06	24.71	100	90.32	38.71	100	94.12	16.18	97.56	87.8	34.15	100	89.16	34.94	100	89.19	13.51	100	83.05	18.64		
	UR	100	99.33	88.33	100	93.33	86.67	100	74.29	68.57	94.74	94.74	89.47	67.57	32.43	12.16	69.44	47.22	27.78	82.14	39.29	10.71	100	84.62	66.67		
	DR	100	96.08	94.12	100	100	100	100	99.01	99.01	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
P7	G	97.44	97.44	97.44	97.46	97.46	97.46	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
	AS	100	87.34	17.72	100	94.81	24.68	100	74.42	4.65	98.39	85.48	3.23	100	89.89	22.47	97.56	86.59	23.17	100	69.44	2.78	98.56	83.61	24.59		
	UR	100	96.67	80	70	100	88	84	100	66.67	66.67	100	85.71	76.19	93.75	81.25	45.83	61.76	32.35	8.82	80	65	35	80	46.67		
	DR	100	100	96.15	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	96.67	
P8	G	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
	AS	100	94.52	15.07	100	89.33	29.33	100	78.57	3.57	100	100	12.96	100	95.15	56.31	100	96.97	63.64	100	94.87	41.03	96.88	90.63	48.44		
	UR	100	96.67	63.33	97.01	85.07	70.15	93.33	80	66.67	97.56	87.8	75.61	72.94	37.65	24.71	52.08	30.21	12.5	92.68	41.46	12.2	98.39	74.19	48.39		
	DR	100	100	93.62	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	

Figure B.2: P1-P8 refers to the 8 participants. Loocomotion modes symbols (G,DR,UR,AS,DS) are compatible with labels given in Table .3. 1, and paths labels are given in Fig. B. 1. Finally, s1,s2 and s3 are pointing to detection accuracy for the first, second and third steps respectively.

3D-Point Cloud dataset of various geometrical terrains in urban environments recorded during human locomotion



C.1 Dataset description

During the evaluation process of the terrain detection algorithm introduced in Chapter 4, we conducted 64 trails of 8 participant in several indoor/outdoor paths with clear and partially occluded path conditions. Posting the raw collected data in on-line dataset could have a potential interest by several scientific communities. Indeed, it can be used by wide range of researchers from different disciplines to improve the control of human locomotion assistive devices (e.g., prosthesis, exo-skeleton, or wheelchairs) using computer vision-based algorithms. In addition, the dataset could be beneficial in implementing vision-based detection systems to improve humanoid robots' navigation over various terrains in urban environments.

Interestingly, UCLouvain provides such opportunity that allows the university researchers to share their data over Dataverse website which is an open source research data repository software. The website provides a unique Digital Objective Identifier for every published dataset DOI (see Fig. C.1).

Each trial consists of three types of recorded data:

- Down-sampled point cloud data: the high density of point cloud data causes processing delay, and not all the points bring a significant part of new information. The point cloud resolution is down sized by the 3D voxel grid filter algorithm, with a leaf size of 4cm (see Fig. C.2). To further reduce the computational cost, a region of interest of one meter wide in front of the user was specified in order to eliminate data lying out of this region.
- Inertial Measurement Unit (IMU): this sensor is built in with the depth camera and it provides instantaneous camera orientation. This piece of information

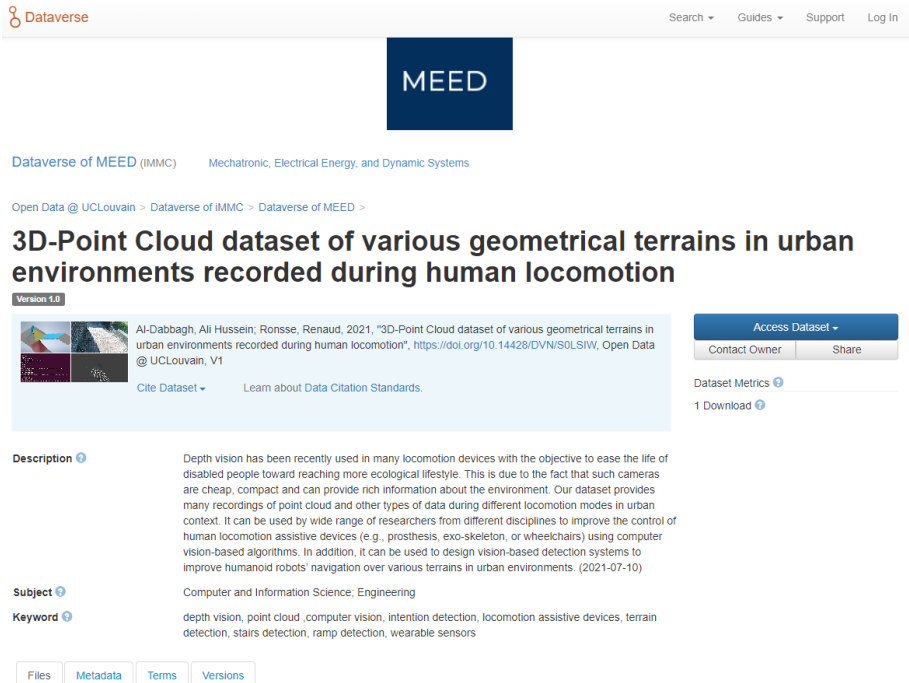


Figure C.1: A screenshot of the published dataset on UCLouvain Dateverse website with DOI: <https://doi.org/10.14428/DVN/S0LSIW>.

can be used for transforming the point cloud data into the actual world frame and obtain kinematic info.

- RGB-image: this is the normal camera image which can be useful to know the path condition and identify the ground truth of each terrain.

The data are recorded with ROS and stored in (*.bag) files. A typical visualization of the recorded data within ROS environment can be seen in Fig. C.3.

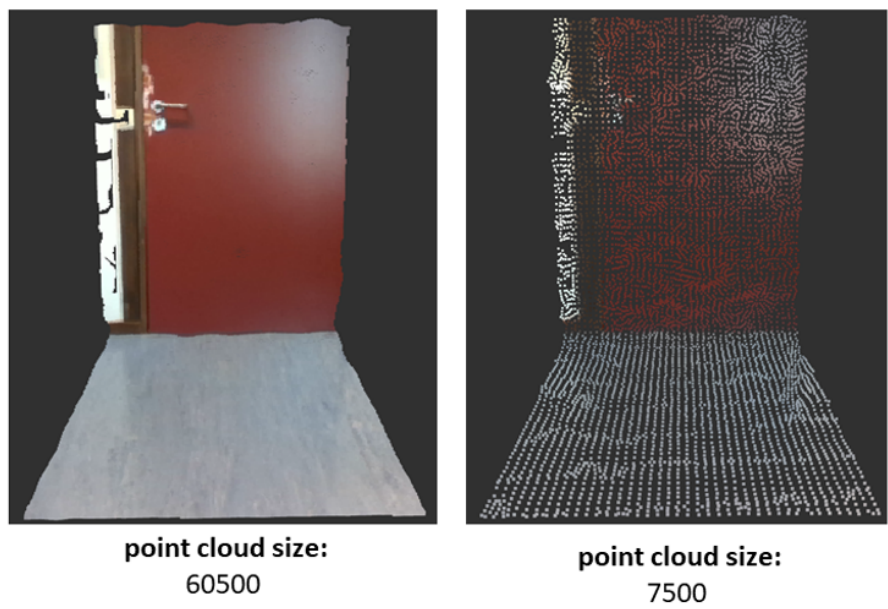


Figure C.2: the left scene is captured with the original point cloud size taken from the camera. The right scene is a downsampled point cloud of the same scene.

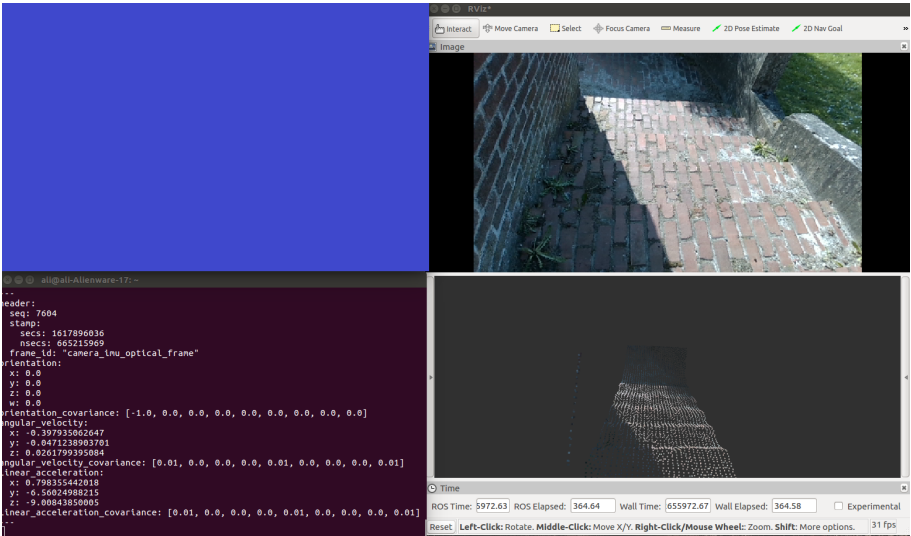


Figure C.3: Typical output of the recorded data. The top window shows an RGB-image of the terrain ahead the user. The bottom left window display an instance of the recorded IMU data of the built in sensor with the camera. The bottom right window display the capture point cloud of the scene in front of the user.

The Algorithm implementation in ROS

D

An important advantage of ROS is that it can run several processing nodes independently and allow interaction between them even if they run different languages. Therefore, it can ease complex system implementation with the possibility of interaction with several sensors. This appendix provides a brief description of the implemented nodes within ROS environment for terrain detection, walking speed, step size and turning angle estimation. The running nodes and their corresponding topics are visualized in Fig. D.1 which is obtained with (rosgraph) command.

The running nodes in the graph are represented in oval shapes while the published topics are represented in rectangular shapes. Starting from the middle of the graph /Terrain_detection is the main running node. In the case of processing recorded data, the node receives the recorded downsampled point cloud and the IMU data to perform affordances detection, while in the case of real time affordance detection, there will be an additional node running that performs downsampling and cropping of the published raw point cloud data by the depth camera. When the down sampled point cloud received, the main node applies several processing step to achieve features extraction and transformation to the world frame with the help of the IMU data which are synchronized with the downsampled point cloud. After that, the node labels the detected planes based on trained SVM classifier decision which runs within /classifier_server node. Then, the locomotion affordances are passed to terrain detection algorithm for multiple steps (still within /Terrain_detection node). In case of ground or ramps affordances, the algorithm relies on the /step_size topic that is instantaneously computed from /Step_size_estimator node based on /Speed_Cadance topic which is published by speed estimator through /rosbridge_websocket node. If stair case is detected, then the algorithm treats each locomotion affordance as single step and order them based on their distance from the user feet. The terrain information for three steps are published under the /TP topic. /Terrain_detection node also publishes point cloud topic named /pointcloud_with_Affordances for visualization purposes. The turning angle is determined within /IMU_Angle node, and the result is published as /IMU_Angle. Finally, the topics related to the terrain information, speed, and

turning angle are all subscribed by `/Path_estimator` node for plotting and visualizing the traveled path.

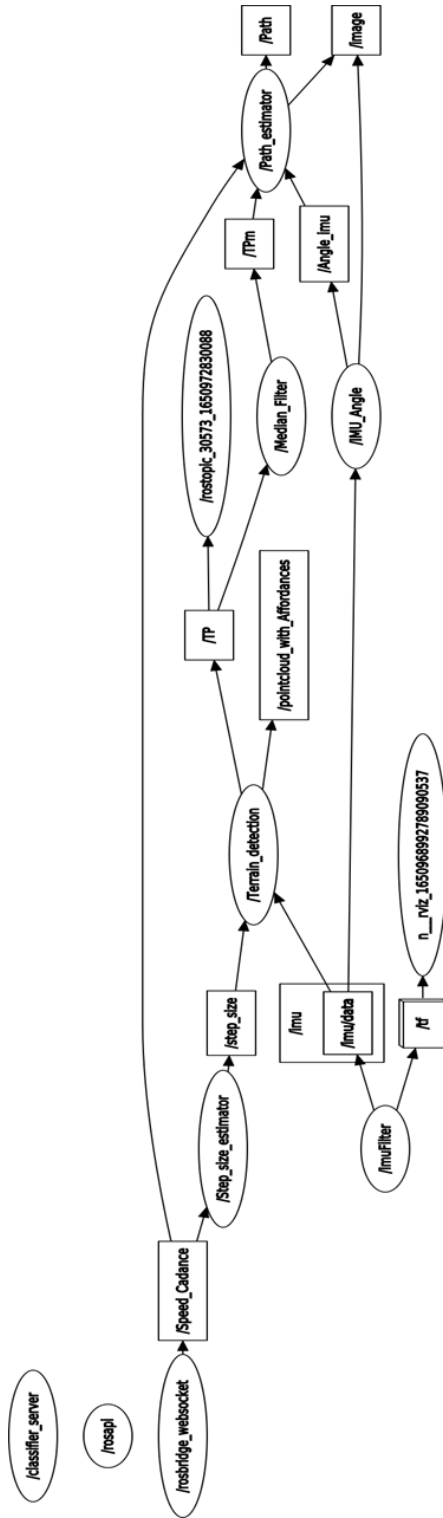


Figure D.1: Block diagram highlights the running nodes and the published topics in ROS environment. Active ROS nodes are displayed in oval shapes while the published information by each node is shown in rectangle shapes.

Bibliography

- [1] D. A. Norman, *The Design of Everyday Things*. New York: Basic Books, 2 ed., 2013.
- [2] A. Kulkarni, D. Chong, and F. A. Batarseh, “Foundations of data imbalance and solutions for a data democracy,” in *data democracy*, pp. 83–106, Elsevier, 2020.
- [3] A. K. Godiyal, V. Verma, N. Khanna, and D. Joshi, “Force myography and its application to human locomotion,” in *Biomedical Signal Processing*, pp. 49–70, Springer, 2020.
- [4] “Multi-annual roadmap for robotics in europe,” 2022.
- [5] R. Riener, M. Rabuffetti, and C. Frigo, “Stair ascent and descent at different inclinations,” *Gait & posture*, vol. 15, no. 1, pp. 32–44, 2002.
- [6] R. Clement, K. E. Bugler, and C. W. Oliver, “Bionic prosthetic hands: A review of present technology and future aspirations,” *The surgeon*, vol. 9, no. 6, pp. 336–340, 2011.
- [7] E. Guizzo and H. Goldstein, “The rise of the body bots [robotic exoskeletons],” *IEEE spectrum*, vol. 42, no. 10, pp. 50–56, 2005.
- [8] M. Goldfarb, B. E. Lawson, and A. H. Shultz, “Realizing the promise of robotic leg prostheses,” *Science translational medicine*, vol. 5, no. 210, pp. 210ps15–210ps15, 2013.
- [9] M. R. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. del R Milán, R. Riener, H. Vallery, and R. Gassert, “Control strategies for active lower extremity prosthetics and orthotics: a review,” *Journal of neuroengineering and rehabilitation*, vol. 12, no. 1, p. 1, 2015.
- [10] O. Christ and P. Beckerle, “Towards active lower limb prosthetic systems: design issues and solutions,” 2016.
- [11] D. Kitwood, “Sophie morgan walks with the aid of ‘rex’, a robotic exoskeleton,” 2022. [Online; accessed June 20, 2022].

- [12] N. Ramstrand and K.-Å. Nilsson, "A comparison of foot placement strategies of transtibial amputees and able-bodied subjects during stair ambulation," *Prosthetics and orthotics international*, vol. 33, no. 4, pp. 348–355, 2009.
- [13] J. Paysant, C. Beyaert, A.-M. Datié, N. Martinet, and A. Jean-Marie, "Influence of terrain on metabolic and temporal gait characteristics of unilateral transtibial amputees," *Journal of Rehabilitation Research and Development*, vol. 43, no. 2, p. 153, 2006.
- [14] D. H. Gates, J. B. Dingwell, S. J. Scott, E. H. Sinitski, and J. M. Wilken, "Gait characteristics of individuals with transtibial amputations walking on a destabilizing rock surface," *Gait & posture*, vol. 36, no. 1, pp. 33–39, 2012.
- [15] F. Heremans, S. Vijayakumar, M. Bouri, B. Dehez, and R. Ronsse, "Bio-inspired design and validation of the efficient lockable spring ankle (elsa) prosthesis," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pp. 411–416, IEEE, 2019.
- [16] M. Tran, L. Gabert, M. Cempini, and T. Lenzi, "A lightweight, efficient fully powered knee prosthesis with actively variable transmission," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1186–1193, 2019.
- [17] L. Flynn, J. Geeroms, R. Jimenez-Fabian, B. Vanderborght, N. Vitiello, and D. Lefeber, "Ankle–knee prosthesis with active ankle and energy transfer: Development of the cyberlegs alpha-prosthesis," *Robotics and Autonomous Systems*, vol. 73, pp. 4–15, 2015.
- [18] L. Najafi and D. Cowan, *Handbook of Electronic Assistive Technology*. Academic Press, 2018.
- [19] Ottobock, "Ottobock back," 2022. [Online; accessed June 20, 2022].
- [20] L. Martin, "Fortis k-srd lockheed martin," 2022. [Online; accessed June 20, 2022].
- [21] D. Novak and R. Riener, "A survey of sensor fusion methods in wearable robotics," *Robotics and Autonomous Systems*, vol. 73, pp. 155–170, 2015.
- [22] A. Parri, K. Yuan, D. Marconi, T. Yan, S. Crea, M. Munih, R. M. Lova, N. Vitiello, and Q. Wang, "Real-time hybrid locomotion mode recognition for lower limb wearable robots," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 6, pp. 2480–2491, 2017.
- [23] L. L. Flynn, J. Geeroms, T. van der Hoeven, B. Vanderborght, and D. Lefeber, "Vub-cyberlegs cybathlon 2016 beta-prosthesis: case study in control of an active two degree of freedom transfemoral prosthesis," *Journal of neuroengineering and rehabilitation*, vol. 15, no. 1, pp. 1–16, 2018.
- [24] J. J. Gibson, *The ecological approach to visual perception: classic edition*. Psychology Press, 2014.

- [25] E. Şahin, M. Cakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, “To afford or not to afford: A new formalization of affordances toward affordance-based robot control,” *Adaptive Behavior*, vol. 15, no. 4, pp. 447–472, 2007.
- [26] A. t. Pas and R. Platt, “Localizing grasp affordances in 3-d points clouds using taubin quadric fitting,” *arXiv preprint arXiv:1311.3192*, 2013.
- [27] P. Kaiser, D. Gonzalez-Aguirre, F. Schültje, J. Borras, N. Vahrenkamp, and T. Asfour, “Extracting whole-body affordances from multimodal exploration,” in *2014 IEEE-RAS International Conference on Humanoid Robots*, pp. 1036–1043, IEEE, 2014.
- [28] P. Kaiser, E. E. Aksoy, M. Grotz, and T. Asfour, “Towards a hierarchy of locomanipulation affordances,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2839–2846, IEEE, 2016.
- [29] G. Lidoris, F. Rohrmuller, D. Wollherr, and M. Buss, “The autonomous city explorer (ace) project—mobile robot navigation in highly populated urban environments,” in *2009 IEEE International Conference on Robotics and Automation*, pp. 1416–1422, IEEE, 2009.
- [30] R. Manduchi, A. Castano, A. Talukder, and L. Matthies, “Obstacle detection and terrain classification for autonomous off-road navigation,” *Autonomous robots*, vol. 18, no. 1, pp. 81–102, 2005.
- [31] I. Halatci, C. A. Brooks, and K. Iagnemma, “Terrain classification and classifier fusion for planetary exploration rovers,” *IEEE Aerospace Conference Proceedings*, 2007.
- [32] L. Iocchi and S. Pellegrini, “Building 3d maps with semantic elements integrating 2d laser, stereo vision and imu on a mobile robot,” in *2nd ISPRS International Workshop 3D-ARCH*, 2007.
- [33] IABA, “Home.”
- [34] Á. Csapó, G. Wersényi, H. Nagy, and T. Stockman, “A survey of assistive technologies and applications for blind users on mobile platforms: a review and foundation for research,” *Journal on Multimodal User Interfaces*, vol. 9, no. 4, pp. 275–286, 2015.
- [35] A. H. Al-dabbagh and R. Ronsse, “A review of terrain detection systems for applications in locomotion assistance,” *Robotics and Autonomous Systems*, p. 103628, 2020.
- [36] B. Kleiner, N. Ziegenspeck, R. Stolyarov, H. Herr, U. Schneider, and A. Verl, “A radar-based terrain mapping approach for stair detection towards enhanced prosthetic foot control,” in *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, pp. 105–110, IEEE, 2018.
- [37] A. H. Al-dabbagh and R. Ronsse, “A new terrain recognition approach for predictive control of assistive devices using depth vision,” in *International Symposium on Wearable Robotics*, pp. 443–447, Springer, 2020.

- [38] B. Zhong, R. L. Da Silva, M. Li, H. Huang, and E. Lobaton, "Environmental context prediction for lower limb prostheses with uncertainty quantification," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 2, pp. 458–470, 2020.
- [39] B. Zhong, R. L. da Silva, M. Tran, H. Huang, and E. Lobaton, "Efficient environmental context prediction for lower limb prostheses," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2021.
- [40] M. F. Eilenberg, H. Geyer, and H. Herr, "Control of a powered ankle–foot prosthesis based on a neuromuscular model," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 18, no. 2, pp. 164–173, 2010.
- [41] B. Kleiner, N. Ziegenspeck, R. Stolyarov, H. Herr, U. Schneider, and A. Verl, "A Radar-Based Terrain Mapping Approach for Stair Detection Towards Enhanced Prosthetic Foot Control," in *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, 2018.
- [42] H. Huang, F. Zhang, L. J. Hargrove, Z. Dou, D. R. Rogers, and K. B. Englehart, "Continuous locomotion-mode identification for prosthetic legs based on neuromuscular–mechanical fusion," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 10, pp. 2867–2875, 2011.
- [43] K. Zhang, C. Xiong, W. Zhang, H. Liu, D. Lai, Y. Rong, and C. Fu, "Environmental Features Recognition for Lower Limb Prostheses Toward Predictive Walking," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 3, pp. 465–476, 2019.
- [44] K. Zhang, J. Luo, W. Xiao, W. Zhang, H. Liu, J. Zhu, Z. Lu, Y. Rong, C. W. de Silva, and C. Fu, "A subvision system for enhancing the environmental adaptability of the powered transfemoral prosthesis," *IEEE Transactions on Cybernetics*, 2020.
- [45] B. Laschowski, W. McNally, A. Wong, and J. McPhee, "Preliminary design of an environment recognition system for controlling robotic lower-limb prostheses and exoskeletons," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pp. 868–873, IEEE, 2019.
- [46] G. Khademi and D. Simon, "Convolutional neural networks for environmentally aware locomotion mode recognition of lower-limb amputees," in *ASME 2019 Dynamic Systems and Control Conference*, American Society of Mechanical Engineers Digital Collection, 2019.
- [47] K. Hauser, "Adaptive locomotion on uneven terrains," *Humanoid Robotics: A Reference*, pp. 1–20, 2016.
- [48] B. Siciliano and O. Khatib, *Springer handbook of robotics*. Springer, 2016.
- [49] N. Ramstrand and K. A. Nilsson, "A comparison of foot placement strategies of transtibial amputees and able-bodied subjects during stair ambulation," *Prosthetics and Orthotics International*, vol. 33, no. 4, pp. 348–355, 2009.

- [50] C. Ferreira, L. P. Reis, and C. P. Santos, "Review of control strategies for lower limb prostheses," in *Robot 2015: Second Iberian Robotics Conference*, pp. 209–220, Springer, 2016.
- [51] H. Vallery, M. R. Tucker, O. Lambercy, R. Gassert, A. Pagel, M. Bouri, J. d. R. Millán, R. Riener, H. Bleuler, and J. Olivier, "Control strategies for active lower extremity prosthetics and orthotics: a review," *Journal of NeuroEngineering and Rehabilitation*, vol. 12, no. 1, p. 1, 2015.
- [52] T. Yan, M. Cempini, C. M. Oddo, and N. Vitiello, "Review of assistive strategies in powered lower-limb orthoses and exoskeletons," *Robotics and Autonomous Systems*, vol. 64, pp. 120–136, 2015.
- [53] J. Lobo-Prat, P. N. Kooren, A. H. Stienen, J. L. Herder, B. F. Koopman, and P. H. Veltink, "Non-invasive control interfaces for intention detection in active movement-assistive devices," *Journal of neuroengineering and rehabilitation*, vol. 11, no. 1, p. 168, 2014.
- [54] A. Parri, E. Martini, J. Geeroms, L. Flynn, G. Pasquini, S. Crea, R. Molino Lova, D. Lefeber, R. Kamnik, M. Munih, *et al.*, "Whole body awareness for controlling a robotic transfemoral prosthesis," *Frontiers in neurorobotics*, vol. 11, p. 25, 2017.
- [55] M. M. Martins, C. P. Santos, A. Frizera-Neto, and R. Ceres, "Assistive mobility devices focusing on smart walkers: Classification and review," *Robotics and Autonomous Systems*, vol. 60, no. 4, pp. 548–562, 2012.
- [56] K. Zhang, C. W. de Silva, and C. Fu, "Sensor fusion for predictive control of human-prosthesis-environment dynamics in assistive walking: A survey," *arXiv preprint arXiv:1903.07674*, 2019.
- [57] B. Hudgins, P. Parker, and R. N. Scott, "A new strategy for multifunction myoelectric control," *IEEE Transactions on Biomedical Engineering*, vol. 40, no. 1, pp. 82–94, 1993.
- [58] A. J. Young, A. Simon, and L. J. Hargrove, "An intent recognition strategy for transfemoral amputee ambulation across different locomotion modes," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pp. 1587–1590, 2013.
- [59] E. Zheng, L. Wang, K. Wei, and Q. Wang, "A noncontact capacitive sensing system for recognizing locomotion modes of transtibial amputees," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 12, pp. 2911–2920, 2014.
- [60] C. Fleischer and G. Hommel, "A human-exoskeleton interface utilizing electromyography," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 872–882, 2008.
- [61] D. Jin, J. Yang, R. Zhang, R. Wang, and J. Zhang, "Terrain identification for prosthetic knees based on electromyographic signal features," *Tsinghua Science and Technology*, vol. 11, no. 1, pp. 74–79, 2006.

- [62] L. J. Hargrove, A. M. Simon, R. Lipschutz, S. B. Finucane, and T. A. Kuiken, "Non-weight-bearing neural control of a powered transfemoral prosthesis," *Journal of NeuroEngineering and Rehabilitation*, vol. 10, no. 1, pp. 1–11, 2013.
- [63] C. D. Hoover, G. D. Fulk, and K. B. Fite, "Stair ascent with a powered transfemoral prosthesis under direct myoelectric control," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 3, pp. 1191–1200, 2013.
- [64] L. Ambrozic, M. Gorsic, J. Geeroms, L. Flynn, R. Molino Lova, R. Kamnik, M. Munih, and N. Vitiello, "CYBERLEGS: A user-oriented robotic transfemoral prosthesis with whole-body awareness control," *IEEE Robotics and Automation Magazine*, vol. 21, no. 4, pp. 82–93, 2014.
- [65] B. Aeyels, W. Van Petegem, J. V. Sloten, G. Van der Perre, and L. Peeraer, "An emg-based finite state approach for a microcomputer-controlled above-knee prosthesis," in *Proceedings of 17th International Conference of the Engineering in Medicine and Biology Society*, vol. 2, pp. 1315–1316, IEEE, 1995.
- [66] S. Au, M. Berniker, and H. Herr, "Powered ankle-foot prosthesis to assist level-ground and stair-descent gaits," *Neural Networks*, vol. 21, no. 4, pp. 654–666, 2008.
- [67] L. Chen, P. Yang, X. Xu, L. Zu, and X. Guo, "Above-knee prosthesis control based on posture recognition by support vector machine," in *2008 IEEE Conference on Robotics, Automation and Mechatronics*, pp. 307–312, IEEE, 2008.
- [68] J. D. Miller, M. S. Beazer, and M. E. Hahn, "Myoelectric walking mode classification for transtibial amputees," *IEEE Transactions on Biomedical engineering*, vol. 60, no. 10, pp. 2745–2750, 2013.
- [69] R. Gupta and R. Agarwal, "Single channel emg-based continuous terrain identification with simple classifier for lower limb prosthesis," *Biocybernetics and Biomedical Engineering*, vol. 39, no. 3, pp. 775–788, 2019.
- [70] K. Englehart, B. Hudgins, P. A. Parker, and M. Stevenson, "Classification of the myoelectric signal using time-frequency based representations," *Medical engineering & physics*, vol. 21, no. 6-7, pp. 431–438, 1999.
- [71] L. J. Hargrove, K. Englehart, and B. Hudgins, "A comparison of surface and intramuscular myoelectric signal classification," *IEEE transactions on biomedical engineering*, vol. 54, no. 5, pp. 847–853, 2007.
- [72] K. Englehart, B. Hudgins, *et al.*, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE transactions on biomedical engineering*, vol. 50, no. 7, pp. 848–854, 2003.
- [73] T. Instruments, "Fdc1004: Basics of capacitive sensing and applications," 2014.
- [74] B. Chen, E. Zheng, X. Fan, T. Liang, Q. Wang, K. Wei, and L. Wang, "Locomotion mode classification using a wearable capacitive sensing system," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 21, no. 5, pp. 744–755, 2013.

- [75] E. Zheng, B. Chen, K. Wei, and Q. Wang, "Lower limb wearable capacitive sensing and its applications to recognizing human gaits," *Sensors*, vol. 13, no. 10, pp. 13334–13355, 2013.
- [76] E. Zheng and Q. Wang, "Noncontact capacitive sensing-based locomotion transition recognition for amputees with robotic transtibial prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 2, pp. 161–170, 2016.
- [77] H. A. Varol, F. Sup, and M. Goldfarb, "Multiclass real-time intent recognition of a powered lower limb prosthesis," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 3, pp. 542–551, 2009.
- [78] B. Chen, E. Zheng, and Q. Wang, "A locomotion intent prediction system based on multi-sensor fusion," *Sensors*, vol. 14, no. 7, pp. 12349–12369, 2014.
- [79] A. J. Young, A. M. Simon, N. P. Fey, and L. J. Hargrove, "Classifying the intent of novel users during human locomotion using powered lower limb prostheses," in *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*, pp. 311–314, IEEE, 2013.
- [80] A. J. Young, A. M. Simon, and L. J. Hargrove, "A training method for locomotion mode prediction using powered lower limb prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 3, pp. 671–677, 2013.
- [81] B. Chen, X. Wang, Y. Huang, K. Wei, and Q. Wang, "A foot-wearable interface for locomotion mode recognition based on discrete contact force distribution," *Mechatronics*, vol. 32, pp. 12–21, 2015.
- [82] S. Wen, F. Wang, and C. Wu, "Realtime gait kinematics classification using lda and svm," in *2011 Chinese Control and Decision Conference (CCDC)*, pp. 592–595, IEEE, 2011.
- [83] H. L. Bartlett and M. Goldfarb, "A phase variable approach for imu-based locomotion activity recognition," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 6, pp. 1330–1338, 2017.
- [84] D. Xu, Y. Feng, J. Mai, and Q. Wang, "Real-Time On-Board Recognition of Continuous Locomotion Modes for Amputees with Robotic Transtibial Prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 10, pp. 2015–2025, 2018.
- [85] H. A. Varol, F. Sup, and M. Goldfarb, "Real-time gait mode intent recognition of a powered knee and ankle prosthesis for standing and walking," in *2008 2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, pp. 66–72, IEEE, 2008.
- [86] A. Mai and S. Commuri, "Intelligent control of a prosthetic ankle joint using gait recognition," *Control Engineering Practice*, vol. 49, pp. 1–13, 2016.

- [87] Mai and S. Commuri, "Gait identification for an intelligent prosthetic foot," in *2011 IEEE International Symposium on Intelligent Control*, pp. 1341–1346, IEEE, 2011.
- [88] M. Islam and E. T. Hsiao-Weckslar, "Detection of gait modes using an artificial neural network during walking with a powered ankle-foot orthosis," *Journal of Biophysics*, vol. 2016, 2016.
- [89] R. Stolyarov, M. Carney, and H. Herr, "Accurate heuristic terrain prediction in powered lower-limb prostheses using onboard sensors," 2020.
- [90] A. J. Young, A. M. Simon, N. P. Fey, and L. J. Hargrove, "Intent recognition in a powered lower limb prosthesis using time history information," *Annals of biomedical engineering*, vol. 42, no. 3, pp. 631–641, 2014.
- [91] Z. Liu, W. Lin, Y. Geng, and P. Yang, "Intent pattern recognition of lower-limb motion based on mechanical sensors," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 651–660, 2017.
- [92] A. J. Young, A. Simon, and L. J. Hargrove, "An intent recognition strategy for transfemoral amputee ambulation across different locomotion modes," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1587–1590, IEEE, 2013.
- [93] A. J. Young and L. J. Hargrove, "A classification method for user-independent intent recognition for transfemoral amputees using powered lower limb prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 2, pp. 217–225, 2015.
- [94] B.-Y. Su, J. Wang, S.-Q. Liu, M. Sheng, J. Jiang, and K. Xiang, "A cnn-based method for intent recognition using inertial measurement units and intelligent lower limb prosthesis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 5, pp. 1032–1042, 2019.
- [95] F. Gao, G. Liu, F. Liang, and W.-H. Liao, "Imu-based locomotion mode identification for transtibial prostheses, orthoses, and exoskeletons," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2020.
- [96] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, 1997.
- [97] R. Ronsse, R. C. Miall, and S. P. Swinnen, "Multisensory integration in dynamical behaviors: maximum likelihood estimation across bimanual skill learning," *Journal of Neuroscience*, vol. 29, no. 26, pp. 8419–8428, 2009.
- [98] H. Huang, S. L. Wolf, and J. He, "Recent developments in biofeedback for neuromotor rehabilitation," *Journal of neuroengineering and rehabilitation*, vol. 3, no. 1, p. 11, 2006.
- [99] R. C. Luo, C.-C. Yih, and K. L. Su, "Multisensor fusion and integration: approaches, applications, and future research directions," *IEEE Sensors journal*, vol. 2, no. 2, pp. 107–119, 2002.

- [100] C. Zhu and W. Sheng, "Human daily activity recognition in robot-assisted living using multi-sensor fusion," in *2009 IEEE International Conference on Robotics and Automation*, pp. 2154–2159, IEEE, 2009.
- [101] F. Zhang and H. Huang, "Real-time recognition of user intent for neural control of artificial legs," in *Proceedings of the 2011 MyoElectric Controls/Powered Prosthetics Symposium Fredericton, Fredericton, NB, Canada*, pp. 14–19, 2011.
- [102] L. Du, F. Zhang, M. Liu, and H. Huang, "Toward design of an environment-aware adaptive locomotion-mode-recognition system," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 10, pp. 2716–2725, 2012.
- [103] X. Zhang, H. Huang, and Q. Yang, "Implementing an fpga system for real-time intent recognition for prosthetic legs," in *Proceedings of the 49th Annual Design Automation Conference*, pp. 169–175, ACM, 2012.
- [104] F. Zhang, W. DiSanto, J. Ren, Z. Dou, Q. Yang, and H. Huang, "A novel cps system for evaluating a neural-machine interface for artificial legs," in *2011 IEEE/ACM Second International Conference on Cyber-Physical Systems*, pp. 67–76, IEEE, 2011.
- [105] R. Hernandez, Q. Yang, H. Huang, F. Zhang, and X. Zhang, "Design and implementation of a low power mobile cpu based embedded system for artificial leg control," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 5769–5772, IEEE, 2013.
- [106] F. Zhang and H. Huang, "Source selection for real-time user intent recognition toward volitional control of artificial legs," *IEEE journal of biomedical and health informatics*, vol. 17, no. 5, pp. 907–914, 2012.
- [107] H. Huang, F. Zhang, L. J. Hargrove, Z. Dou, D. R. Rogers, and K. B. Englehart, "Continuous locomotion-mode identification for prosthetic legs based on neuromuscular - Mechanical fusion," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 10 PART 1, pp. 2867–2875, 2011.
- [108] I.-K. Jung and S. Lacroix, "High resolution terrain mapping using low altitude aerial stereo imagery,," in *ICCV*, vol. 3, p. 946, 2003.
- [109] S. A. Bouhamed, I. K. Kallel, and D. S. Masmoudi, "Stair case detection and recognition using ultrasonic signal," in *2013 36th International Conference on Telecommunications and Signal Processing (TSP)*, pp. 672–676, IEEE, 2013.
- [110] M. Liu, D. Wang, and H. Helen Huang, "Development of an Environment-Aware Locomotion Mode Recognition System for Powered Lower Limb Prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 4, pp. 434–443, 2016.
- [111] F. Zhang, Z. Fang, M. Liu, and H. Huang, "Preliminary design of a terrain recognition system," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5452–5455, IEEE, 2011.

- [112] S. Carvalho, J. Figueiredo, and C. P. Santos, "Environment-aware locomotion mode transition prediction system," in *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pp. 1–6, IEEE, 2019.
- [113] X. Zhang, D. Wang, Q. Yang, and H. Huang, "An automatic and user-driven training method for locomotion mode recognition for artificial leg control," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 6116–6119, IEEE, 2012.
- [114] M. A. Richards, *Fundamentals of Radar Signal Processing*. US: McGraw-Hill.
- [115] C. Panagiotakis and A. Argyros, "Parameter-free modelling of 2D shapes with ellipses," *Pattern Recognition*, vol. 53, pp. 259–275, 2016.
- [116] S. Sahoo, M. Maheshwari, D. K. Pratihari, and S. Mukhopadhyay, "A geometry recognition-based strategy for locomotion transitions' early prediction of prosthetic devices," *IEEE Transactions on Instrumentation and Measurement*, 2019.
- [117] K. Ishiwata, M. Sekiguchi, M. Fuchida, and A. Nakamura, "Basic study on step detection system for the visually impaired," in *2013 IEEE International Conference on Mechatronics and Automation*, pp. 1332–1337, IEEE, 2013.
- [118] S. Al Mamun, A. Lam, Y. Kobayashi, and Y. Kuno, "Single laser bidirectional sensing for robotic wheelchair step detection and measurement," in *International Conference on Intelligent Computing*, pp. 37–47, Springer, 2017.
- [119] A. Pérez-Yus, G. López-Nicolás, and J. J. Guerrero, "Detection and modelling of staircases using a wearable depth sensor," in *European Conference on Computer Vision*, pp. 449–463, Springer, 2014.
- [120] G. Capi and H. Toda, "A new robotic system to assist visually impaired people," in *2011 RO-MAN*, pp. 259–263, IEEE, 2011.
- [121] Z. Zheng, G. Zhong, and H. Deng, "A method to detect stairs with three-dimensional scanning for hexapod robot stair climbing," in *2016 IEEE International Conference on Mechatronics and Automation*, pp. 2541–2546, IEEE, 2016.
- [122] S. Oßwald, J.-S. Gutmann, A. Hornung, and M. Bennewitz, "From 3d point clouds to climbing stairs: A comparison of plane segmentation approaches for humanoids," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*, pp. 93–98, IEEE, 2011.
- [123] T. Westfechtel, K. Ohno, B. Mertsching, D. Nickchen, S. Kojima, and S. Tadokoro, "3d graph based stairway detection and localization for mobile robots," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 473–479, IEEE, 2016.
- [124] L. Nalpantidis, V. Krüger, J. O. Eklundh, and A. Gasteratos, "Object Detection and Terrain Classification in Agricultural Fields Using 3D Lidar Data," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9163, pp. 188–197, 2015.

- [125] M. Bibuli, M. Caccia, and L. Lapierre, "Vision-aided Guidance and Navigation for Close Formation Flight," *Journal of Field Robotics*, vol. 7, no. PART 1, pp. 81–86, 2015.
- [126] M. Hebert, "Active and passive range sensing for robotics," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 1, pp. 102–110, IEEE, 2000.
- [127] B. Langmann, K. Hartmann, and O. Loffeld, "Depth camera technology comparison and performance evaluation.," in *ICPRAM (2)*, pp. 438–444, Citeseer, 2012.
- [128] J. Fu, S. Wang, Y. Lu, S. Li, and W. Zeng, "Kinect-like depth denoising," in *2012 IEEE International Symposium on Circuits and Systems*, pp. 512–515, IEEE, 2012.
- [129] T. Braun, H. Bitsch, and K. Berns, "Visual terrain traversability estimation using a combined slope/elevation model," in *Annual Conference on Artificial Intelligence*, pp. 177–184, Springer, 2008.
- [130] À. Santamaria-Navarro, E. H. Teniente, M. Morta, and J. Andrade-Cetto, "Terrain classification in complex three-dimensional outdoor environments," *Journal of Field Robotics*, vol. 32, no. 1, pp. 42–60, 2015.
- [131] Y. H. Lee, T.-S. Leung, and G. Medioni, "Real-time staircase detection from a wearable stereo system," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 3770–3773, IEEE, 2012.
- [132] T. J. J. Tang, W. L. D. Lui, and W. H. Li, "Plane-based detection of staircases using inverse depth," in *Australasian conference on robotics and automation (ACRA)*, 2012.
- [133] S. Caraiman, A. Morar, M. Owczarek, A. Burlacu, D. Rzeszutarski, N. Botezatu, P. Herghelegiu, F. Moldoveanu, P. Strumillo, and A. Moldoveanu, "Computer vision for the visually impaired: the sound of vision system," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1480–1489, 2017.
- [134] H. Harms, E. Rehder, T. Schwarze, and M. Lauer, "Detection of ascending stairs using stereo vision," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2496–2502, IEEE, 2015.
- [135] H. A. Varol and Y. Massalin, "A feasibility study of depth image based intent recognition for lower limb prostheses," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 5055–5058, IEEE, 2016.
- [136] Y. Massalin, M. Abdrakhmanova, and H. A. Varol, "User-independent intent recognition for lower limb prostheses using depth sensing," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 8, pp. 1759–1770, 2017.

- [137] A. Perez-Yus, D. Gutiérrez-Gómez, G. Lopez-Nicolas, and J. Guerrero, "Stairs detection with odometry-aided traversal from a wearable rgb-d camera," *Computer Vision and Image Understanding*, vol. 154, pp. 192–205, 2017.
- [138] P. Costa, H. Fernandes, P. Martins, J. Barroso, and L. J. Hadjileontiadis, "Obstacle detection using stereo imaging to assist the navigation of visually impaired people," *Procedia Computer Science*, vol. 14, pp. 83–93, 2012.
- [139] M. Nixon and A. S. Aguado, *Feature extraction and image processing for computer vision*. Academic Press, 2012.
- [140] S. Wang, H. Pan, C. Zhang, and Y. Tian, "Rgb-d image-based detection of stairs, pedestrian crosswalks and traffic signs," *Journal of Visual Communication and Image Representation*, vol. 25, no. 2, pp. 263–272, 2014.
- [141] S. Wang and Y. Tian, "Detecting stairs and pedestrian crosswalks for the blind by rgb-d camera," in *2012 IEEE International Conference on Bioinformatics and Biomedicine Workshops*, pp. 732–739, IEEE, 2012.
- [142] S. Murakami, M. Shimakawa, K. Kivota, and T. Kato, "Study on stairs detection using rgb-depth images," in *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*, pp. 1186–1191, IEEE, 2014.
- [143] R. Munoz, X. Rong, and Y. Tian, "Depth-aware indoor staircase detection and recognition for the visually impaired," in *2016 IEEE international conference on multimedia & expo workshops (ICMEW)*, pp. 1–6, IEEE, 2016.
- [144] D. S. Chan, R. K. Silva, J. C. Monteiro, and F. Lizarralde, "Efficient stairway detection and modeling for autonomous robot climbing," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5916–5921, IEEE, 2017.
- [145] X. Lu and R. Manduchi, "Detection and localization of curbs and stairways using stereo vision," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 4648–4654, IEEE, 2005.
- [146] J. A. Delmerico, D. Baran, P. David, J. Ryde, and J. J. Corso, "Ascending stairway modeling from dense depth imagery for traversability analysis," in *2013 IEEE International Conference on Robotics and Automation*, pp. 2283–2290, IEEE, 2013.
- [147] P. V. Hough, "Method and means for recognizing complex patterns," Dec. 18 1962. US Patent 3,069,654.
- [148] T. Yan, Y. Sun, T. Liu, C.-H. Cheung, and M. Q.-H. Meng, "A locomotion recognition system using depth images," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6766–6772, IEEE, 2018.
- [149] M. Bellone, G. Reina, N. I. Giannoccaro, and L. Spedicato, "Unevenness point descriptor for terrain analysis in mobile robot applications," *International Journal of Advanced Robotic Systems*, vol. 10, no. 7, p. 284, 2013.

- [150] M. Bibuli, M. Caccia, and L. Lapierre, "Terrain Classification in Complex Three-dimensional Outdoor Environments," *Journal of Field Robotics*, vol. 7, no. PART 1, pp. 81–86, 2015.
- [151] K. Yang, K. Wang, R. Cheng, and X. Zhu, "A new approach of point cloud processing and scene segmentation for guiding the visually impaired," 2015.
- [152] R. B. Rusu and S. Cousins, "Point cloud library (pcl)," in *2011 IEEE international conference on robotics and automation*, pp. 1–4, 2011.
- [153] L. Souto, A. Castro, L. Gonçalves, and T. Nascimento, "Stairs and doors recognition as natural landmarks based on clouds of 3d edge-points from rgb-d sensors for mobile robot localization," *Sensors*, vol. 17, no. 8, p. 1824, 2017.
- [154] M. Vlaminck, L. Jovanov, P. Van Hese, B. Goossens, W. Philips, and A. Pižurica, "Obstacle detection for pedestrians with a visual impairment based on 3d imaging," in *2013 International Conference on 3D Imaging*, pp. 1–7, IEEE, 2013.
- [155] H.-H. Pham, T.-L. Le, and N. Vuillerme, "Real-time obstacle detection system in indoor environment for the visually impaired using microsoft kinect sensor," *Journal of Sensors*, vol. 2016, 2016.
- [156] N. E. Krausz, T. Lenzi, and L. J. Hargrove, "Depth sensing for improved control of lower limb prostheses," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 11, pp. 2576–2587, 2015.
- [157] A. Ciobanu, A. Morar, F. Moldoveanu, L. Petrescu, O. Ferche, and A. Moldoveanu, "Real-time indoor staircase detection on mobile devices," in *2017 21st International Conference on Control Systems and Computer Science (CSCS)*, pp. 287–293, IEEE, 2017.
- [158] A. Dargazany and K. Berns, "Terrain traversability analysis using organized point cloud, superpixel surface normals-based segmentation and pca-based classification," in *Workshop on field and assistive robotics, Lahore, Pakistan*, pp. 17–18, 2014.
- [159] D. Holz, S. Holzer, R. B. Rusu, and S. Behnke, "Real-time plane segmentation using rgb-d cameras," in *Robot Soccer World Cup*, pp. 306–317, Springer, 2011.
- [160] M. Bellone, A. Messina, and G. Reina, "A new approach for terrain analysis in mobile robot applications," in *2013 IEEE International Conference on Mechatronics (ICM)*, pp. 225–230, IEEE, 2013.
- [161] G. Reina, M. Bellone, L. Spedicato, and N. I. Giannoccaro, "3d traversability awareness for rough terrain mobile robots," *Sensor Review*, vol. 34, no. 2, pp. 220–232, 2014.
- [162] R. S. Consensus, "A paradigm for model fitting with applications to image analysis and automated cartography," *MA Fischler, RC Bolles*, vol. 6, pp. 381–395, 1981.

- [163] L. Li, F. Yang, H. Zhu, D. Li, Y. Li, and L. Tang, "An improved ransac for 3d point cloud plane segmentation based on normal distribution transformation cells," *Remote Sensing*, vol. 9, no. 5, p. 433, 2017.
- [164] K. Yang, K. Wang, R. Cheng, W. Hu, X. Huang, and J. Bai, "Detecting traversable area and water hazards for the visually impaired with a prgb-d sensor," *Sensors*, vol. 17, no. 8, p. 1890, 2017.
- [165] H.-C. Huang, C.-T. Hsieh, and C.-H. Yeh, "An indoor obstacle detection system using depth information and region growth," *Sensors*, vol. 15, no. 10, pp. 27116–27141, 2015.
- [166] M. Nejati and B. D. Argall, "Automated incline detection for assistive powered wheelchairs," in *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 1007–1012, IEEE, 2016.
- [167] K. Zhang, W. Zhang, W. Xiao, H. Liu, C. W. de Silva, and C. Fu, "Sequential decision fusion for environmental classification in assistive walking," *arXiv preprint arXiv:1904.11152*, 2019.
- [168] S. Kuthirummal, A. Das, and S. Samarasekera, "A graph traversal based algorithm for obstacle detection using lidar or stereo," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3874–3880, IEEE, 2011.
- [169] A. Sinha, P. Papadakis, and M. R. Elara, "A staircase detection method for 3d point clouds," in *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*, pp. 652–656, IEEE, 2014.
- [170] H. Yue, W. Chen, X. Wu, and J. Zhang, "Kinect based real time obstacle detection for legged robots in complex environments," in *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, pp. 205–210, IEEE, 2013.
- [171] D. Belter, P. Łabecki, P. Fankhauser, and R. Siegwart, "Rgb-d terrain perception and dense mapping for legged robots," *International Journal of Applied Mathematics and Computer Science*, vol. 26, no. 1, pp. 81–97, 2016.
- [172] N. E. Krausz and L. J. Hargrove, "Recognition of ascending stairs from 2D images for control of powered lower limb prostheses," *International IEEE/EMBS Conference on Neural Engineering, NER*, vol. 2015-July, no. Figure 2, pp. 615–618, 2015.
- [173] K. Du, J. Xin, Y. Shi, D. Liu, and Y. Zhang, "A high-precision vision-based mobile robot slope detection method in unknown environment," in *2018 Chinese Automation Congress (CAC)*, pp. 3192–3197, IEEE.
- [174] A. Cherian, V. Morellas, and N. Papanikolopoulos, "Accurate 3D ground plane estimation from a single image," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2243–2249, 2009.
- [175] D. C. Hernandez and K.-H. Jo, "Stairway segmentation using gabor filter and vanishing point," in *2011 IEEE International Conference on Mechatronics and Automation*, pp. 1027–1032, IEEE, 2011.

- [176] I. Bogoslavskyi, O. Vysotska, J. Serafin, G. Grisetti, and C. Stachniss, "Efficient traversability analysis for mobile robots using the kinect sensor," in *2013 European Conference on Mobile Robots*, pp. 158–163, IEEE, 2013.
- [177] T. Vllasolli, B. Zafirova, N. Orovcaneć, A. Poposka, A. Murtezani, and B. Krasniqi, "Energy expenditure and walking speed in lower limb amputees: a cross sectional study.," *Ortopedia, traumatologia, rehabilitacija*, vol. 16, no. 4, pp. 419–426, 2014.
- [178] V. Ruiz Garate, A. Parri, T. Yan, M. Munih, R. Molino Lova, N. Vitiello, and R. Ronsse, "Experimental validation of motor primitive-based control for leg exoskeletons during continuous multi-locomotion tasks," *Frontiers in neuro-robotics*, vol. 11, p. 15, 2017.
- [179] U. Gregory and L. Ren, "Intent prediction of multi-axial ankle motion using limited emg signals," *Frontiers in Bioengineering and Biotechnology*, vol. 7, 2019.
- [180] X.-F. Han, J. S. Jin, M.-J. Wang, W. Jiang, L. Gao, and L. Xiao, "A review of algorithms for filtering the 3d point cloud," *Signal Processing: Image Communication*, vol. 57, pp. 103–112, 2017.
- [181] K. Mirzaei, M. Arashpour, E. Asadi, H. Masoumi, Y. Bai, and A. Behnood, "3d point cloud data processing with machine learning for construction and infrastructure applications: A comprehensive review," *Advanced Engineering Informatics*, vol. 51, p. 101501, 2022.
- [182] A. Nguyen and B. Le, "3d point cloud segmentation: A survey," in *2013 6th IEEE conference on robotics, automation and mechatronics (RAM)*, pp. 225–230, IEEE, 2013.
- [183] T. Rabbani, F. Van Den Heuvel, and G. Vosselmann, "Segmentation of point clouds using smoothness constraint," *International archives of photogrammetry, remote sensing and spatial information sciences*, vol. 36, no. 5, pp. 248–253, 2006.
- [184] E. Grilli, F. Menna, and F. Remondino, "A review of point clouds segmentation and classification algorithms," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 42, p. 339, 2017.
- [185] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [186] A. H. A. Al-dabbagh and R. Ronsse, "Using depth vision for terrain detection during active locomotion," in *International Conference on Intelligent Robots and Systems (IROS)*, pp. 485–492, IEEE, 2021.
- [187] J. S. Rodriguez, "A comparison of an rgb-d cameras performance and a stereo camera in relation to object recognition and spatial position determination," *ELCVIA Electronic Letters on Computer Vision and Image Analysis*, vol. 20, no. 1, pp. 16–27, 2021.

- [188] A. Kadambi, A. Bhandari, and R. Raskar, “3d depth cameras in vision: Benefits and limitations of the hardware,” in *Computer vision and machine learning with RGB-D sensors*, pp. 3–26, Springer, 2014.
- [189] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng, *et al.*, “Ros: an open-source robot operating system,” in *ICRA workshop on open source software*, vol. 3, p. 5, Kobe, Japan, 2009.
- [190] R. B. Rusu and S. Cousins, “3d is here: Point cloud library (pcl),” in *2011 IEEE international conference on robotics and automation*, pp. 1–4, IEEE, 2011.
- [191] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Pasos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [192] A. H. Al-Dabbagh and R. Ronsse, “3D-Point Cloud dataset of various geometrical terrains in urban environments recorded during human locomotion,” 2021.
- [193] J.-H. Kim, “Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap,” *Computational statistics & data analysis*, vol. 53, no. 11, pp. 3735–3745, 2009.
- [194] D. Berrar, “Cross-validation,” in *Encyclopedia of Bioinformatics and Computational Biology* (S. Ranganathan, M. Gribskov, K. Nakai, and C. Schönbach, eds.), pp. 542–545, Oxford: Academic Press, 2019.
- [195] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, “Scikit-learn: Machine learning in python,” *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.
- [196] Y. Qian, Y. Wang, C. Chen, J. Xiong, Y. Leng, H. Yu, and C. Fu, “Predictive locomotion mode recognition and accurate gait phase estimation for hip exoskeleton on various terrains,” *IEEE Robotics and Automation Letters*, 2022.
- [197] A. Mital, H. Fard, and H. Khaledi, “A biomechanical evaluation of staircase riser heights and tread depths during stair-climbing,” *Clinical Biomechanics*, vol. 2, no. 3, pp. 162–164, 1987.
- [198] P. R. Cavanagh, R. Kram, *et al.*, “Stride length in distance running: velocity, body dimensions, and added mass effects,” *Med Sci Sports Exerc*, vol. 21, no. 4, pp. 467–79, 1989.
- [199] C. Pew and G. K. Klute, “Turn intent detection for control of a lower limb prosthesis,” *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 4, pp. 789–796, 2017.
- [200] K. Brzostowski, “Novel approach to human walking speed enhancement based on drift estimation,” *Biomedical Signal Processing and Control*, vol. 42, pp. 18–29, 2018.

- [201] Z. He and W. Zhang, "Estimation of walking speed using accelerometer and artificial neural networks," in *International Workshop on Computer Science for Environmental Engineering and EcoInformatics*, pp. 42–47, Springer, 2011.
- [202] G. Fusco and J. M. Coughlan, "Indoor localization using computer vision and visual-inertial odometry," in *International conference on computers helping people with special needs*, pp. 86–93, Springer, 2018.
- [203] Z. Guo, Q. Yu, R. Guo, and H. Lu, "Structural features based visual odometry for indoor textureless environments," in *2020 Chinese Automation Congress (CAC)*, pp. 3984–3989, IEEE, 2020.
- [204] H. Darbandi, F. Serra Bragança, B. J. Van der Zwaag, J. Voskamp, A. I. Gmel, E. H. Haraldsdóttir, and P. Havinga, "Using different combinations of body-mounted imu sensors to estimate speed of horses—a machine learning approach," *Sensors*, vol. 21, no. 3, p. 798, 2021.
- [205] Y. MejiaCruz, J. Franco, G. Hainline, S. Fritz, Z. Jiang, J. M. Caicedo, B. Davis, and V. Hirth, "Walking speed measurement technology: a review," *Current Geriatrics Reports*, vol. 10, no. 1, pp. 32–41, 2021.
- [206] J. Yoon, H.-S. Park, and D. L. Damiano, "A novel walking speed estimation scheme and its application to treadmill control for gait rehabilitation," *Journal of neuroengineering and rehabilitation*, vol. 9, no. 1, pp. 1–13, 2012.
- [207] S. Yang and Q. Li, "Inertial sensor-based methods in walking speed estimation: A systematic review," *Sensors*, vol. 12, no. 5, pp. 6102–6116, 2012.
- [208] S. Lei, A. Honglei, G. Jialong, M. Hongxu, and W. Qing, "Real-time walking speed estimation by involving phase variables based on deep learning," in *2021 China Automation Congress (CAC)*, pp. 2839–2844, IEEE, 2021.
- [209] T.-N. Do, R. Liu, C. Yuen, M. Zhang, and U.-X. Tan, "Personal dead reckoning using imu mounted on upper torso and inverted pendulum model," *IEEE Sensors Journal*, vol. 16, no. 21, pp. 7600–7608, 2016.
- [210] J.-S. Hu, K.-C. Sun, and C.-Y. Cheng, "A kinematic human-walking model for the normal-gait-speed estimation using tri-axial acceleration signals at waist location," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 8, pp. 2271–2279, 2013.
- [211] J.-S. Hu, K.-C. Sun, and C.-Y. Cheng, "A model-based human walking speed estimation using body acceleration data," in *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1985–1990, IEEE, 2012.
- [212] K. Brzostowski, "Toward the unaided estimation of human walking speed based on sparse modeling," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 6, pp. 1389–1398, 2018.
- [213] S. Zhu, H. Anderson, and Y. Wang, "A real-time on-chip algorithm for imu-based gait measurement," in *Pacific-Rim Conference on Multimedia*, pp. 93–104, Springer, 2012.

- [214] X. Meng, Z.-Q. Zhang, J.-K. Wu, W.-C. Wong, and H. Yu, "Self-contained pedestrian tracking during normal walking using an inertial/magnetic sensor module," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 3, pp. 892–899, 2013.
- [215] H. Vathsangam, A. Emken, D. Spruijt-Metz, and G. S. Sukhatme, "Toward free-living walking speed estimation using gaussian process-based regression with on-body accelerometers and gyroscopes," in *2010 4th International Conference on Pervasive Computing Technologies for Healthcare*, pp. 1–8, IEEE, 2010.
- [216] S. Zihajehzadeh and E. J. Park, "A gaussian process regression model for walking speed estimation using a head-worn imu," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2345–2348, IEEE, 2017.
- [217] Y. Song, S. Shin, S. Kim, D. Lee, and K. H. Lee, "Speed estimation from a tri-axial accelerometer using neural networks," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3224–3227, IEEE, 2007.
- [218] J.-g. Park, A. Patel, D. Curtis, S. Teller, and J. Ledlie, "Online pose classification and walking speed estimation using handheld devices," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pp. 113–122, 2012.
- [219] A. Paraschiv-Ionescu, C. J. Newman, L. Carcreff, C. N. Gerber, S. Armand, and K. Aminian, "Locomotion and cadence detection using a single trunk-fixed accelerometer: validity for children with cerebral palsy in daily life-like conditions," *Journal of neuroengineering and rehabilitation*, vol. 16, no. 1, pp. 1–11, 2019.
- [220] I. Corporation, "How-to: Getting imu data from d435i and t265."
- [221] S. Zihajehzadeh and E. J. Park, "Experimental evaluation of regression model-based walking speed estimation using lower body-mounted imu," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 243–246, IEEE, 2016.
- [222] C. Tudor-Locke, E. J. Aguiar, H. Han, S. W. Ducharme, J. M. Schuna, T. V. Barreira, C. C. Moore, M. A. Busa, J. Lim, J. R. Sirard, *et al.*, "Walking cadence (steps/min) and intensity in 21–40 year olds: Cadence-adults," *International Journal of Behavioral Nutrition and Physical Activity*, vol. 16, no. 1, pp. 1–11, 2019.
- [223] C. Tudor-Locke, S. W. Ducharme, E. J. Aguiar, J. M. Schuna, T. V. Barreira, C. C. Moore, C. J. Chase, Z. R. Gould, M. A. Amalbert-Birriel, J. Mora-Gonzalez, *et al.*, "Walking cadence (steps/min) and intensity in 41 to 60-year-old adults: the cadence-adults study," *International Journal of Behavioral Nutrition and Physical Activity*, vol. 17, no. 1, pp. 1–10, 2020.
- [224] M. P. Murray, "Walking patterns of normal women," *Arch Phys Med Rehabil*, vol. 51, pp. 637–650, 1970.

- [225] M. P. Murray, A. B. Drought, and R. C. Kory, "Walking patterns of normal men," *JBJS*, vol. 46, no. 2, pp. 335–360, 1964.
- [226] Y. C. Han, K. I. Wong, and I. Murray, "Stride length estimation based on a single shank's gyroscope," *IEEE Sensors Letters*, vol. 3, no. 10, pp. 1–4, 2019.
- [227] J.-D. Sui and T.-S. Chang, "Imu based deep stride length estimation with self-supervised learning," *IEEE Sensors Journal*, vol. 21, no. 6, pp. 7380–7387, 2021.
- [228] C. Lu, H. Uchiyama, D. Thomas, A. Shimada, and R.-i. Taniguchi, "Indoor positioning system based on chest-mounted imu," *Sensors*, vol. 19, no. 2, p. 420, 2019.

