POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Development and validation of a trip-eliciting system: towards trip detection and fall avoidance for lower limb amputees

MASTER THESIS IN BIOMEDICAL ENGINEERING

Pittoni Lorenzo 10572396

Advisor:
Alessandra Laura Giulia Pedrocchi

Co-advisors:
Silvestro Micera

Academic year:
2022-2023

Abstract: **Trips constitute a significant health risk for several subject typologies. Among them, lower limb prosthesis users are particularly susceptible to such perturbations, as their condition makes it difficult to avoid falls. Active prostheses could aid in reducing the fall risk, but to do so they need to be capable of identifying trips as they happen. To develop this capability, a database with data on the dynamics of trips needs to be created. Therefore, a device capable of deploying obstacles was designed and used to cause trips during validation trials. Sensors were placed on the subjects, acquiring a preliminary database containing dynamic data from the tripping events. Analysis of the data generated confirmed that the setup was capable of reliably causing realistic trips. Furthermore, the data collected resulted promising for the future development of a detection algorithm.**

Key-words: trip, stumble, lower limb prosthesis, gait analysis, trip detection, fall risk reduction

1. Introduction

1.1 Incidence of trips and falls in lower limb prosthesis users

Lower limb prosthesis (LLP) users report falling at least once a year, with an incidence of more than 50% [1]. A fall for an LLP user can result in injuries, like fractures, sprains, or bruises. These injuries lead to physical pain and discomfort and may require medical attention, rehabilitation, and prolonged recovery periods. This constitutes an additional source of expenses for the patients and for the healthcare system [2].

According to a study on the topic [3], approximately 50% of LLP users experience fear of falling (FoF), regardless of whether they have recently experienced a fall (although those who have fallen report fear with a greater incidence). Among those who experience FoF, 75% reported that they avoided activity due to their fear. LLP users may become anxious about engaging in activities that they perceive as risky or challenging. This fear can significantly limit their mobility and independence, negatively affecting their quality of life.

Activity avoidance can result in decreased physical activity levels and sedentary behavior, leading to a decline in overall health and fitness. Avoiding activities that place the user at an increased fall risk can be considered as an adaptive behavior to their new limitations. This behavior however becomes maladaptive when it results in over-restriction. At this point inactivity can further contribute to muscle weakness, reduced cardiovascular fitness, and compromised balance, increasing the risk of future stumbling and falls [4].

The combination of all these factors can significantly impact the overall quality of life for LLP users. They may experience limitations in their ability to perform daily tasks, engage in social interactions, and participate in hobbies or recreational activities. This can lead to feelings of isolation, frustration, and a decreased sense of well-being.

As one in five falls can be attributed to a trip, and as 40% of falls can be attributed either to a slip or a stumble[1], reducing the probability that one such event results in a fall would go a long way in improving the quality of life of LLP users. A possible solution could be to develop an active prosthesis capable of reacting to a stumble or a slip and executing a successful recovery. Presently active swing control strategies predominantly treat the swing phase motion as an “open loop” problem with respect to trip hazards [5], meaning no corrective action is taken.

One of the modules of INAIL’s MOTU++ project was dedicated to developing a prosthesis that had this capability, using sensors that could be placed on board the prosthesis. In order to do so an algorithm capable of correctly identifying a stumble or a slip event is necessary, which in turn requires data from tripping events. My work in the frame of the project commissioned to the University of Sant’Anna revolved around the development and validation of this experimental setup, which I will describe in this work.

1.2 The trip: cause and recovery

In the framework of this project, we will use the terms “trip” and “stumble” interchangeably to refer to the same kind of event: a perturbation applied during the swing phase of the gait to the swinging foot. In everyday life, this kind of perturbation is usually elicited by impact with an obstacle and may result in a fall, if the recovery strategies implemented fail.

Although a tripping event can happen in a plethora of situations, it is possible to use the minimum toe clearance to predict the likelihood of stumbling during gait. The minimum toe clearance (MTC) is the minimum distance between the foot and the ground when the foot reaches its peak forward velocity during swing. The theory suggests that a small MTC combined with greater variability in toe clearance height raises the risk of the swing foot making unintended contact with an unseen obstacle, the kind of event that initiates a stumble [6]. MTC is susceptible to the ankle angle, and as lower limb amputees lack ankle dorsiflexion muscles they have a smaller MTC. While it is possible for LLP users to implement compensatory strategies to increase the MTC, these result in an increase in energy consumption while walking. These two reasons combined increase the likelihood of trips for LLP users. Prosthetic designs that increase the MTC are then particularly effective in reducing

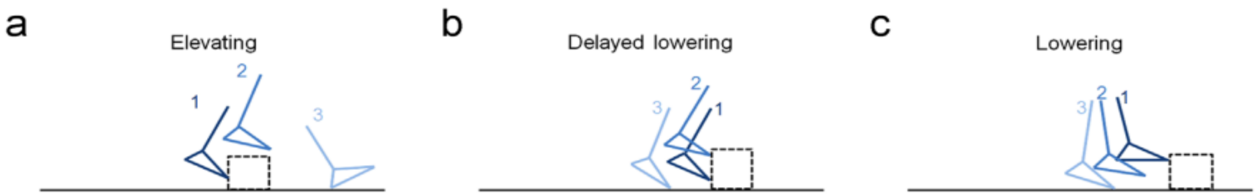


Figure 1, diagram representing the three recovery strategies: (a) elevating, (b) delayed-lowering, and (c) lowering strategy.

the trip probability, like active dorsiflexing prosthesis [7], [8], though they don't necessarily implement solutions to avoid falls.

After a tripping event, three standard recovery strategies are expected: elevating strategy (Figure 1a), lowering strategy (Figure 1b), and delayed lowering strategy (Figure 1c) [7]. In the first case, the foot is directly lifted above the obstacle, while in the second case, the foot of the swinging leg is quickly lowered, shortening the step. The third strategy is the result of a failed attempt at elevating the foot above the obstacle, which causes the patient to lower the foot in the end, terminating the step early. The choice of strategy appears to be correlated with the gait phase in which the trip happens [8]. A trip during the early swing results in the employment of the elevating and the delayed lowering strategy, while a trip in the late swing requires the lowering strategy. During the mid-swing, either of these strategies can be employed.

Amputation modifies the neuromuscular system by eliminating both the direct sensory input from the amputated limb, as well as the direct control of the motor output of the distal limb segments. This considerably affects the gait of LLP users and the tripping recovery strategy. It was indeed found that LLP users employ two additional responses to tripping events: the hopping strategy and the skipping strategy [9]. For the first strategy, the amputee attempts recovery by jumping over the obstacle with both legs, while for the second, an extra step is taken with the tripped foot before moving the support foot. The skipping strategy was employed only when the trip happened on the sound side while hopping was mostly performed for a trip in the early swing on the prosthetic side.

1.3 Methods for gait analysis

The human gait can be characterized through several parameters of interest. Depending on the application we may be interested in stride velocity, joint angle, or swing time, to cite a few. Usually from these parameters, it is possible to evaluate the impairment level of a patient. For example, poststroke hemiparetic gait is characterized by asymmetry, which can be measured by swing time, stance time, or an intralimb ratio of swing to stance time [10]. Different methods and techniques have been developed to measure these parameters, starting with semi-subjective analysis techniques. These techniques consist of analyses carried out in clinical conditions by a specialist. The patient's various gait-related parameters are observed and evaluated while they walk on a pre-determined circuit. Although these techniques have the advantage of not requiring special equipment to be carried out, the subjective nature of the evaluation affects the accuracy, exactitude, repeatability, and reproducibility of the measurements.

Nowadays it is possible to conduct objective analysis using different devices to capture and measure information related to the various gait parameters. These methods can be divided into three categories: those based on image acquisition (IA), on floor sensors (FS), and on sensors located on the body, carried by the users (wearable sensors, WS).

The typical IA system is formed by several digital or analog cameras with lenses that can be used to gather gait-related information. Depending on the system, markers are placed on the subject to track the body segment, and after the acquisition the image data is processed through specialized software to obtain the trajectories of the various markers. This kind of system can provide useful information on the position of body segments and on their speed and acceleration, although these have to be derived from position. On the other side calibration, post-processing time, and the necessity of specialized equipment constitute a significant drawback for this kind of technique. Furthermore, marker placement can suffer from operator-related errors and partially affect the mobility of the subject. Markerless motion capture methods do exist and they offer an alternative to these problems, but their lower accuracy compared to the other methods has limited their medical application.

In FS systems, sensors are placed along the floor on instrumented walkways where gait is measured by pressure or force sensors and moment transducers when the subject walks on them. There are two types of floor sensors: force platforms and pressure measurement systems. The first can be used to directly measure the force vector applied, which is not possible with pressure measurement systems. These are useful for quantifying the pressure patterns under a foot over time but cannot quantify horizontal or shear components of the applied forces. These systems can be particularly useful to detect specific time instants and distinguish phases in the gait. The information that can be retrieved from these systems is limited though, as the only source of information is the interaction between the subject and the floor.

Finally, WS systems involve the placement of sensors on the patient's body. These kinds of systems allow us to perform a variety of measurements, as different sensors can be employed. Force and pressure measurements are possible through sensors integrated into instrumented shoes and soles that are then donned by the patients. Inertial sensors are electronic devices that measure and report on an object's velocity, acceleration, orientation, and gravitational forces, using a combination of accelerometers and gyroscopes, and sometimes magnetometers. These types of sensors may be fitted within an IMU device (Inertial Measurement Unit). Electrogoniometers can be used to study the angles of ankles, knees, hips, and metatarsals. The electromyogram (EMG) signal can be retrieved through the use of electrodes and can provide useful information on muscle activation during gait. WS systems guarantee the most flexibility, thanks to the modularity of the sensors and lacking the necessity of specialized environments, and cumbersome equipment. At the same time though they can affect the movement of the patient, which in turn can affect the final results of the measurements.

1.4 Experimental setups for tripping

As stated before, trips can lead to serious health complications if they result in a fall. It doesn't surprise then that several experimental setups have been devised to study this phenomenon. Unfortunately, there is a significant lack of standardization for what concerns the definition of a trip, the method for trip inducement, the details recorded, and the method of analysis. For this reason, it is difficult to compare the different experimental setups, although a rough categorization based on trip inducement methodology is still possible. Three main techniques have been identified: tether-based perturbations (1.4.1) [8] consisting of a tether or rope being attached to the participants' ankles becoming taut during the swing phase; a treadmill-based method (1.4.2) consisting of sudden accelerations and decelerations through gait [11]; and an obstacle being manually or automatically actuated (1.4.3) [12] to halt the progression of the foot through the mid-swing phase.

The protocol for this kind of experiment usually involves the subject walking on a predetermined path or on a treadmill. While walking a perturbation is unexpectedly supplied to the subject, which results in the end of the trial, depending on the setup. For treadmill- and tether-based setups usually the experiment keeps going until the desired amount of tripping events have been recorded, keeping the uncertainty by randomizing the time between perturbations. For object-based setups, each tripping event is associated with one trial. Generally speaking, adaptation is considered a negative factor when collecting data on tripping events, as expecting a perturbation can affect gait characteristics. For this reason, techniques to distract the subject or cover noise from the tripping devices are employed, like having the subject perform the experiment in a double-task configuration [13], earphones with white noise, or hidden or movable obstacles [14]. Furthermore, for setups where a single obstacle in a fixed position was employed, only the first trial was analyzed, as knowing the placement of the obstacle was considered to have too much influence on the subsequent trial. For what concerns data acquisition, most setups collected data from either accelerometers or markers placed on the leg of the subject, or both, depending on the experiment's purpose.

1.4.1 Tether-based perturbation

As previously stated, tether-based perturbation employs one or two tethers with one end attached to the patient foot or feet. Pulling the tether connected to the swinging foot can elicit a perturbation, to which the body will respond by employing the same strategies employed in a trip with a physical obstacle [9]. This kind of operation can be even performed by simply having a human operator pull on the tether [15] or controlling the tension via a switch [16]. This introduces some complications though since it becomes impossible to associate a time stamp automatically and accurately with the tripping event. In paper [15] this was solved by identifying the time instant associated with the fall and analyzing a time window before it, but this of course requires the patient to fall.

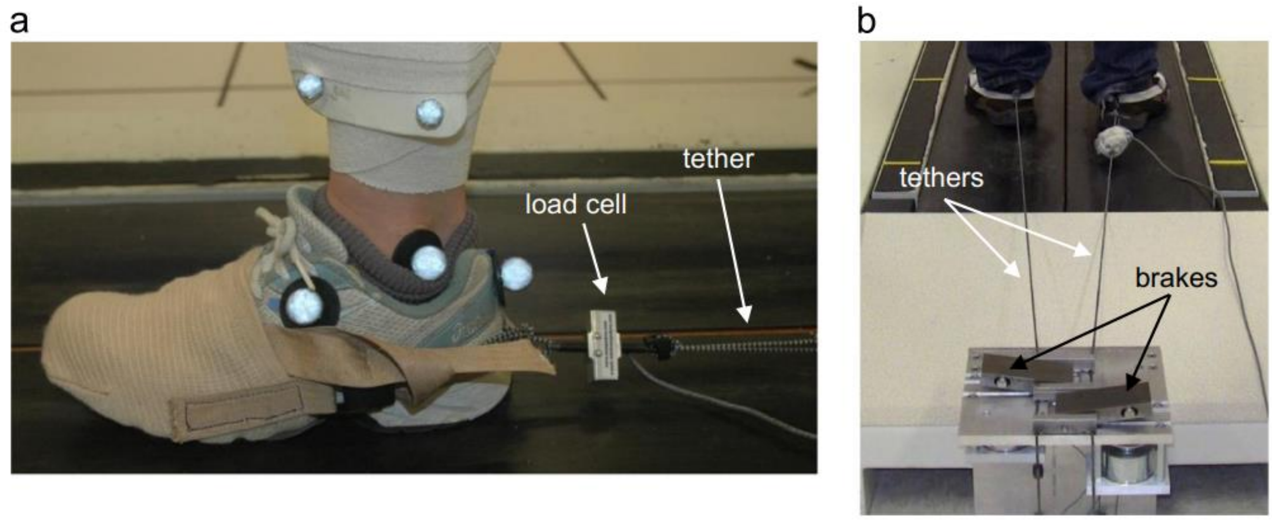


Figure 2, tripping setup developed in paper [8] with (a) two tethers attached to each of the subject feet and (b) solenoid-driven brakes to arrest the movement of the tethers

Although it requires a more complex setup, it is possible to implement an automatic tripping device based on tethers. In this case is possible to obtain a time stamp associated with the tripping, simplifying the data analysis. Automatic tripping setups employ treadmills [8], [17] since tether length is finite, so they force patients to walk at a fixed speed during the experiment. This simplifies the tripping algorithm, as there is less variability in swing duration for the same patient, but the resulting gait is less similar to the natural one. An example of this kind of setup is shown in Figure 2.

This technique presents one major drawback in correlation to our project. While it is true that the response of the patient is the same in the case of tether tripping and physical object tripping[8], we cannot know for certain if the accelerations to which the tripping leg is subjected are equivalent. First of all, the way the tether is attached to the foot influences the direction of application of the forces, which may result in a different direction with respect to a real-life trip. Even if the direction were the same, the block and release of the tether may introduce artifacts. Finally, none of the studies employing tethers performed measurements on the acceleration to which the leg is subjected, meaning there is no way to compare the performance of the system with a real-life trip. The reason for this is that most of the studies that we found were concerned with the response to the tripping event. Only in the paper [15], there was interest in identifying a tripping event, but it was done through the kinematic analysis of the whole fall event. Since the goal of our project is to eventually develop an algorithm capable of identifying the tripping event as it happens, we need to be sure that our system causes the patient's leg to experience acceleration close to real-life conditions.

1.4.2 Treadmill-based perturbation

In treadmill-based perturbation, a sudden acceleration or deceleration of the treadmill belt is used to unbalance the patient. This can be done while the patient is standing on the treadmill [18] or while they are walking [11], [19], [20]. Depending on how the perturbation is applied, it is possible to use this technique to simulate a trip or a slip [20]. Considering a walking patient, a trip is generally caused by a sudden acceleration of the treadmill, applied during the swing phase of the leg we desire to trip. This means that the perturbation is applied to the stance leg, but the strategies employed by the patient to recover from the event are comparable to that of a real-life trip [18].

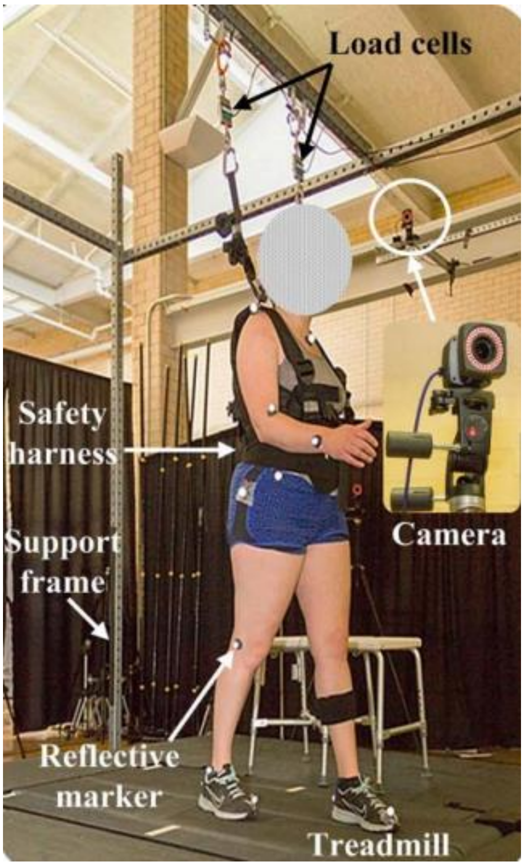


Figure 3, example for experimental setup using a treadmill, presented in paper [19]

For this kind of setup, both single-belt and dual-belt treadmills have been used, in conjunction with devices capable of extracting gait characteristics, such as force or pressure plates, or simply force-sensitive insoles. This allows the system to automatically supply the perturbation when the patient finds themselves in the right instant of the gait. For example, a slip could be elicited by decelerating the belt during the double support phase.

Unfortunately, there are some drawbacks when using this strategy to develop a trip detection algorithm. First of all, using a treadmill introduces consistencies in walking that may reduce the false alarm rate, since it forces the patient to walk at a fixed speed. More importantly, a tripping perturbation produces passive changes in the kinematics and kinetics of the perturbed limb [7]. These are due to the impact of the foot with the obstacle, which results in knee and ankle rotations. As such, the real kinematic situation can be quite different from the one simulated by using a treadmill [20]. Considering this, it may not be possible to effectively develop a trip detection algorithm based on acceleration when using this technique.

1.4.3 Obstacle-based perturbation

This kind of technique is probably the one that presents the highest variability among different experimental setups. Techniques mostly differ in object deployment strategy, which can be divided into passive, manually deployed, and automatic deployment. In the first case, the patients are made to walk on an uneven surface and trips occur randomly when the patients' feet get caught in the surface [13], [21], [22]. In the other two cases, the most common kind of obstacle is represented by a spring-actuated bar standing up or emerging from the ground [14], [23]–[27]. These obstacles are usually locked in their loaded position and their locking mechanism is actuated by a solenoid. Depending on the kind of system, the release of the obstacle can be decided by an operator or automatically performed by an algorithm. This kind of setup can still involve a treadmill, although in that case the obstacle is usually a block dropped at the top of the treadmill that reaches the tripping foot as the treadmill moves [28], an example of which is visible in Figure 4.

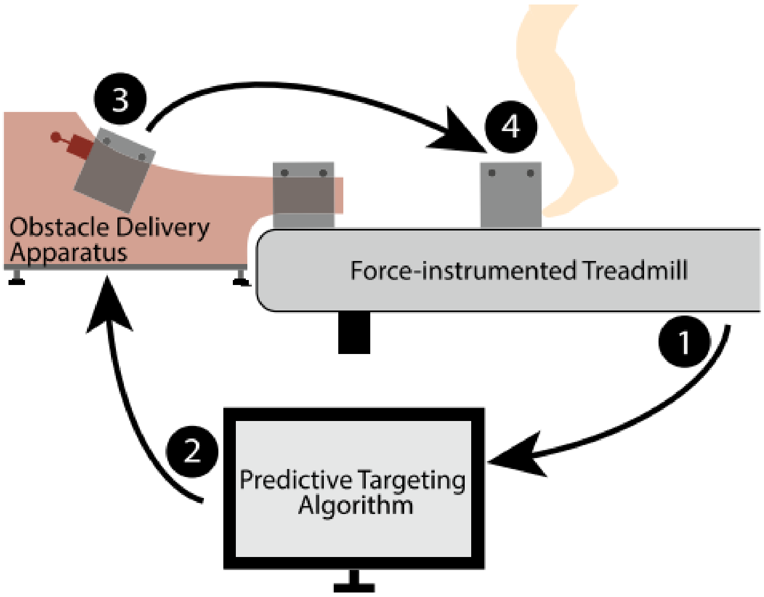


Figure 4, schematic of a tripping setup where the obstacle is delivered onto a treadmill, developed in paper [28]

Obstacles appeared to be the most promising approach for this project, as in daily life trips verify upon the impact of the foot with an unseen obstacle. For this motive, it is reasonable to expect that an experimental setup involving an obstacle will generate dynamic data comparable to real-life data. Unfortunately, these setups are the most complex to implement among those presented up until now. Firstly, the deployment of the obstacle requires time that affects the possibility of selecting a specific phase of the gait; then, in the case of spring-actuated systems, the release mechanism can fail or introduce further delay in the deployment of the obstacle; finally, the rigidity and the compliance of the obstacle need to be tuned, as the obstacle needs to alter the trajectory of the foot without fully stopping it.

Passive and manually deployed obstacle setups are easier to develop since there is no need to automatically identify an instant for the activation of the obstacle. At the same time though, there are some relevant drawbacks. For passive systems, there is no way of knowing when the subject impacted an obstacle. This could be solved by instrumenting each obstacle with strain sensors, but this would exponentially increase the complexity of the system, given the high volume of obstacles. The only viable solution would be to align the recording of the trial with the data collected, but this would require extensive manual post-processing of the signal. Manually deployed obstacle setups do allow associating automatically a time stamp to the tripping event, but they are less reliable. It is up to the operator to decide when to deploy the obstacle and having a human in the control loop introduces some latencies and variability. Since adaptability is an issue when simulating trips, it is important to limit the number of trials performed, meaning failed trials should be kept to a minimum. Automatically deployed obstacles have the advantages of associating a time stamp to the tripping event and having an overall high reliability. They are significantly more complex to implement though, since different sensors need to communicate between themselves. At the same time, the complexity mainly affects the development time rather than the feasibility of implementing this kind of system.

2. Materials and methods

2.1 The concept

As stated in the introduction (1.1), one of the objectives of the MOTU++ project was the development of a trip detection algorithm that used accelerometers placed on board the prosthesis. To collect the data necessary for the development of the algorithm, we devised an experimental setup capable of causing trips and collecting dynamic data on the tripping leg. Since the algorithm needs to be effective in real-life applications, it needs to be developed from data similar to real-life situations. For this reason, the trips needed to be as life-like as possible. This introduced some requirements: first of all, the forces acting on the leg should be comparable to those acting in a real trip; the gait should be as close as possible to the regular gait of the subject; the trip needs to be unexpected, to avoid anticipatory movements that would alter the gait. We also decided to introduce another requirement: the likelihood of a fall should be kept to a minimum. This additional requirement came from the fact that we were not interested in studying the falling aspect of the tripping event, but only the onset of the perturbation. Furthermore, a fall constitutes a considerable health hazard, especially for fragile subjects.

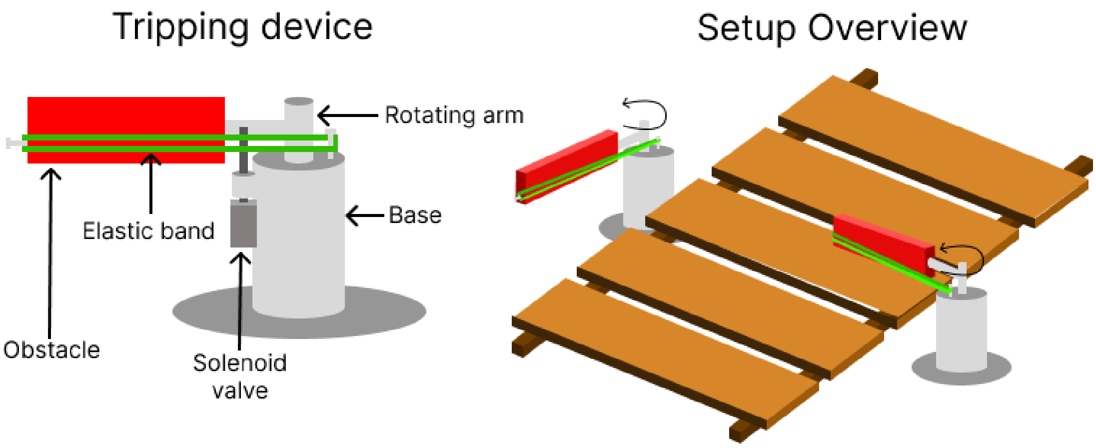


Figure 5, schematic representation of the tripping setup: on the left, the individual tripping device; on the right, the overall setup, with the walkway and the obstacle attached to the side.

To meet the first requirement, we decided that the tripping perturbation should be caused by the automatic deployment of a physical obstacle. Among the techniques analyzed, this appeared the most promising for simulating a realistic tripping event from the dynamic point of view. The second requirement meant that we could not use a treadmill, since this would have artificially regularized the gait. Concerning the additional requirement we introduced, we decided that the obstacle would have to be compliant, in order not to completely stunt the progression of the foot. Furthermore, having a compliant obstacle meant that we could also characterize trips with obstacles like rugs or branches, which are not fixed to the ground. The subjects were also secured with a harness supported by a winch, to eliminate the risk of falls.

Excluding the treadmill meant that the obstacle had to be spring-actuated. To utilize this kind of obstacle, we created a walkway on top of which the subject could walk. This gave us a solid structure for fastening the tripping devices and a clear path where the subject could walk, facilitating the alignment between the tripping foot and the obstacle. The obstacle, visible on the left in Figure 5, is a rotating bar. While loaded, the obstacle bar stays parallel to the longer side of the walkway, and when it is released it becomes perpendicular to the direction the foot is moving in, resulting in an impact. The instant of activation is determined from two sources: the gait phase, obtained from sensors placed under the tripping foot of the subject; and the subject position on the walkway, retrieved by distance sensors. When the subject's tripping foot is in the right position with respect to the obstacle and in the correct gait phase, the obstacle is released and the tripping foot is obstructed, resulting in a trip. The flow diagram for the tripping setup is presented in Figure 6.

Since the activation of a single tripping device requires some specific conditions to verify, to increase the chances of activation more than one obstacle was placed along the walkway, as exemplified by the scheme in Figure 5. This also served another purpose: this kind of obstacle would have been particularly difficult to conceal, so having multiple obstacles that could potentially activate hampered the capacity of the subject to predict when and where the tripping attempt would happen. Furthermore, the strict requirement for activation introduced a certain factor of randomization to the trial, making it more difficult for the subject to know whether the obstacle would be released in their next step. This fulfilled the last requirement of the system that we had yet to tackle.

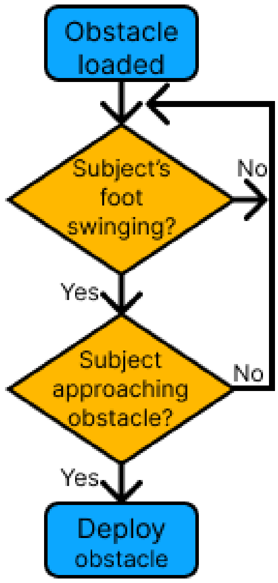


Figure 6, flow diagram the control system for the tripping setup.

2.2 Hardware

The hardware comprises all the physical elements necessary for the proper working of the experimental setup, be it mechanical or electronic in nature.

2.2.1 Walkway and Obstacles

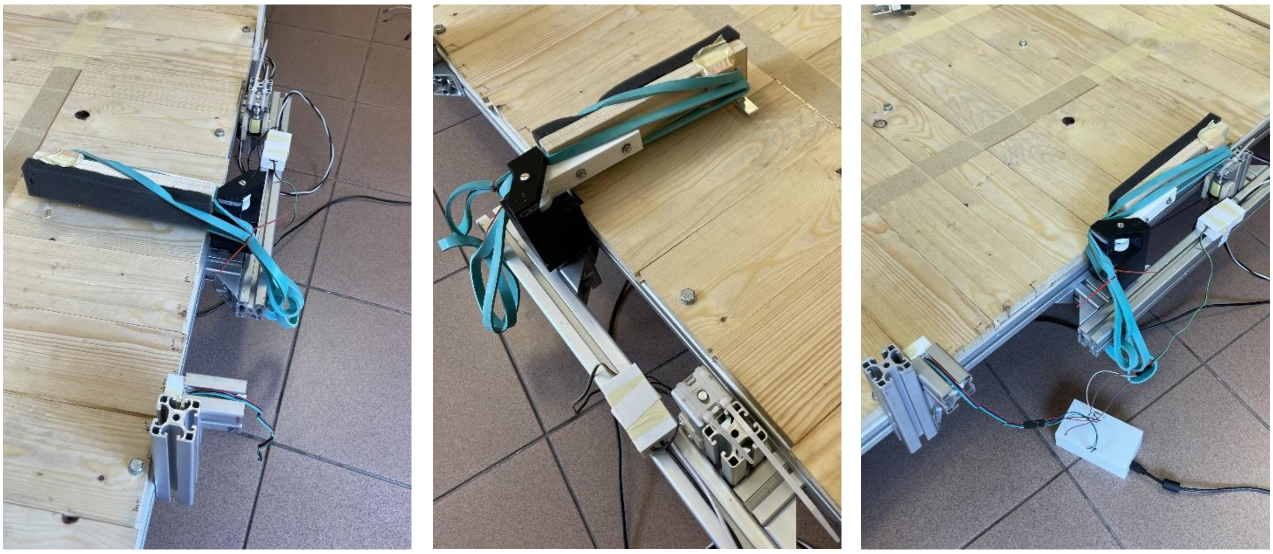
The walkway was obtained by mounting 2-meter-long wooden boards on an aluminum frame, whose dimensions were 2 meters by 3 meters. The aluminum frame had 15-centimeters-high legs, resulting in an elevated platform. This gave us further clearance for attaching the obstacles and running cables for the microprocessor employed. This structure constituted a single module of the walkway, which in the final iteration would comprise 3 modules, for a total length of 9 meters. The validation setup included only two modules as it is shown in Figure 7.

The obstacles had 3 main elements: the base, the rotating arm, and the obstacle proper. The base and the rotating arm were connected by a screw passing through both. This was the pivot around which the rotating arm could rotate. To reduce attrition between the screw and the arm, they were not directly connected: two ball bearings act as a junction between the two of them, meaning the rotating arm could freely rotate around the screw. Finally, the obstacle was fixed to the arm using two screws. The screws were placed far away from each other, to prevent the obstacle from rotating around the junction and to better distribute the strain applied to the rotating arm during actuation.

Figure 7, the walkway.

The base and the rotating arm were both 3D printed out of PLA. For the base, the choice of material was made to simplify manufacturing: this way we could rapidly produce a replacement or update the design of the element. For the rotating part, the PLA was chosen because it is light, which guarantees a faster deployment of the obstacle. The main drawback of using PLA was its strength: since the forces acting on the device were fairly high, more than once these elements broke, meaning the design had to be adjusted to increase their resistance. Since the obstacle was the element subjected to the highest solicitations, it was instead obtained from plywood, which is sturdier, although not as light as PLA. For what concerns its dimensions, the obstacle was 2-centimeter-thick, with a length of 20 centimeters and a height of 5 centimeters. The length was selected to guarantee only one of the feet would be affected, while the height chosen meant that the foot would impact with the obstacle without remaining caught with it. To make sure that the impact with the obstacle wouldn’t be painful, the plywood board was covered with 3 centimeters of foam. The obstacle is visible from different angles in Figure 8.

To actuate each obstacle, elastic bands were employed. This allowed us to make compliant obstacles, the strength of which we could also regulate to adapt them to different kinds of subjects. To regulate their strength, their length was adjusted: there needed to be enough strength to obtain a fast rotation and an obstacle stiff enough to cause a stumble; at the same time, too much tension in the band increased the force necessary to unlock the obstacle, putting the release of the locking mechanism in jeopardy. For the final configuration, we selected an elastic length of 22 cm, which meant the release strength had to be 17 Newton. The elastic band was connected to the base on one side and to the distal end of the plywood board on the other. The configuration of the band can be seen in Figure 8.



(a) Deployed obstacle, front view (b) Deployed obstacle, back view (c) Loaded obstacle

Figure 8, the obstacle from different angles.

Finally, a metal hook was fixed to the distal end of the plywood board. The hook was an L-shaped metal strip with a hole in the middle of one of the segments, with a thickness of a couple of millimeters. To lock the obstacle in the loaded position, a metal pin capable only of vertical movement was lifted through the hole. To release the obstacle, the pin had to be pulled downwards. The pulling was performed by a solenoid, for reasons that will be better explained in the actuators section (2.2.3). This choice introduced some issues though: the pulling strength of a solenoid sharply decreases the more the moving element, called a plunger, is extracted from the coil that houses it. Furthermore, the force decreases if the plunger loses its alignment with the coil. The first issue meant that we had to arrange the hook and the locking mechanism in order to minimize the lifting of the locking pin. Then, to prevent misalignment, the pin was made to pass through a metal block with a hole in it, fixed to the walkway. This stopped the pin from wiggling around when the loaded obstacle was pulling on it. These two adjustments allowed the solenoid to express maximum pulling strength. Since most of the force requirements for the release came from attrition between the pin and the hook, we also decided to lubricate the junction to further facilitate the release. The final configuration of the obstacle and the locking mechanism is presented schematically in **Error! Reference source not found.**.

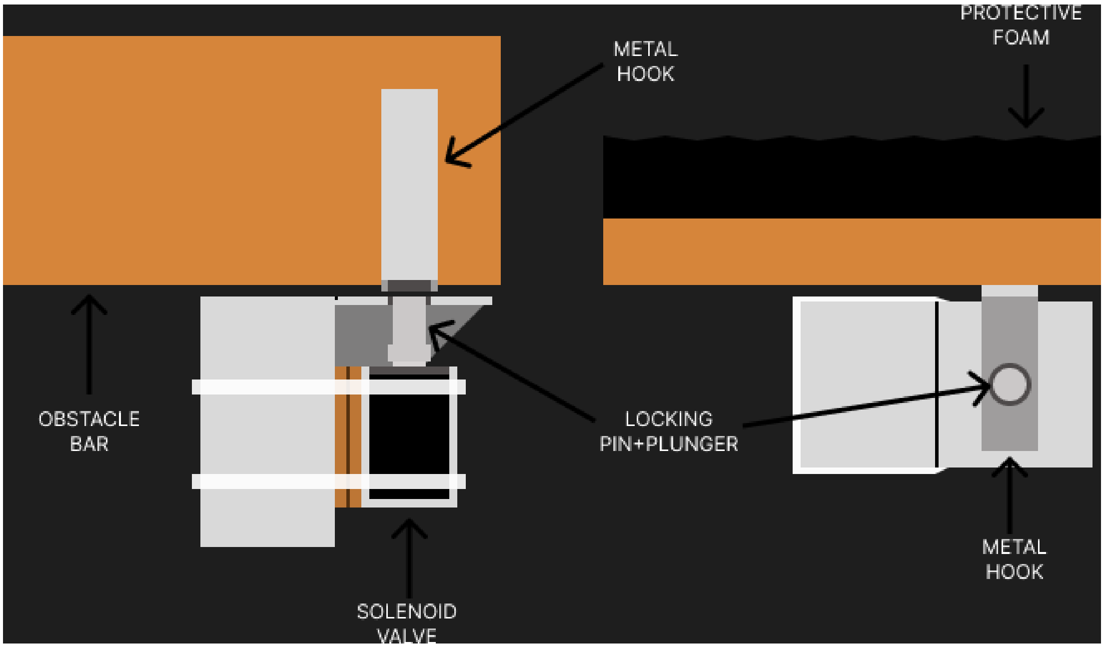


Figure 9, locking mechanism schematic, with focus on the mechanical elements. Two views of the mechanism are presented: from the back on the left and from the top on the right.

2.2.2 Sensors

As stated in the previous section, the obstacle needs to be deployed when the subject is close enough to the obstacle and the tripping foot is in the swing phase. The first information can correspond to a trigger sent when the subject reaches a certain position in relation to the obstacle. To do this we decided to use a distance sensor placed on the side of the platform. The main reason for this choice was that we wanted a way to discriminate which leg would trigger the device: a distance sensor would allow us to identify whether the passing leg was the support or the tripped one since the second would be the closest to the side of the platform (and therefore to the sensor). Figure 10 shows how this is implemented in our system. Distance sensors either work using ultrasounds or lasers to perform the measurements, in both cases sending a signal out and waiting for it to return. In this case, we selected a Time-of-Flight distance sensor, the VL53L0X, which obtains the distance as a function of the time it took the laser sent by the sensor to bounce onto something and return. The choice was made since laser-based distance sensors have a greater sensitivity, allowing for easier discrimination between the legs. Laser-based sensors have also a much higher sampling frequency, and we were under strict time requirements when operating the system in real-time. The sensor was mounted on a custom PCB, which allowed us to connect the sensor to a microcontroller unit.

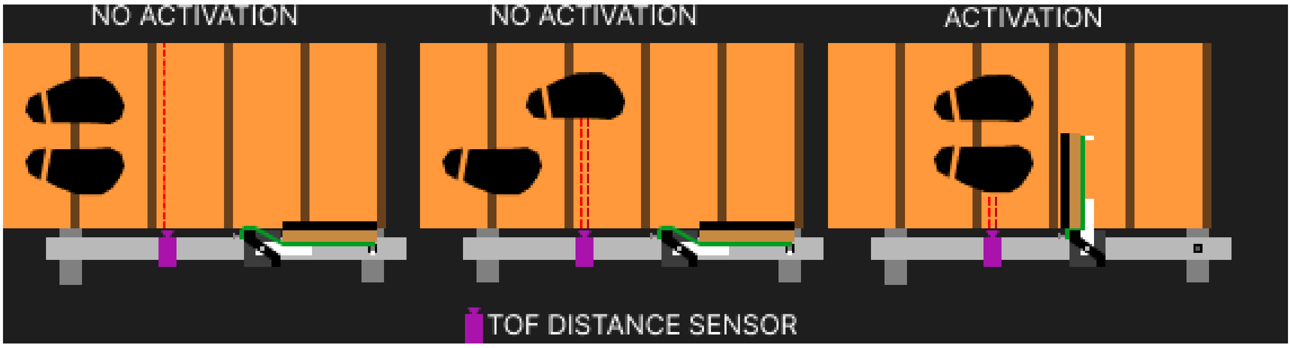


Figure 10, different behaviors of the system depending on the distance detected.

A major drawback of the sensors selected was that, when no obstacle was detected inside its working range, the output returned randomly jumped to zero. We tried to solve this by placing wooden planks on the opposite side of the walkway with respect to the sensor. Two issues still remained though: reading started to become unreliable if no obstacle was detected even a meter from the obstacle, which was much shorter than the working range of two meters; alignment between the plank and the sensors was difficult because we only had an approximate idea of the laser path. To fully solve this issue filtering was deemed necessary and the implementation of this solution will be described in the software section (2.3.1).

The other information necessary for the correct deployment of the obstacle was the gait phase. For each foot, there is a swing phase and a stance phase. The first begins when the foot detaches from the ground, in the instant of toe-off, and it ends when contact with the ground is made again, in the instant of heel strike. When the swing phase ends, the stance phase begins, and vice versa. Therefore, to identify the phases we needed to identify those two time instants. We can define them in terms of load: since in the swing phase no load is applied to the swinging foot, heel strike corresponds to a transition from having no load on the heel to a loaded heel; toe-off corresponds instead to the instant when the load at the tip of the foot is removed. The natural choice for identifying these instants is therefore some kind of pressure or force sensor placed under the shoe. The sensors we decided to employ were Force Sensing Resistors (FSR). These have the advantage of being thin and flexible, making them easy to place and unencumbering for the subject. For the rest of the project, we will refer to the FSRs employed as Footswitches (FS).

Figure 11, an example of FSR, similar
to the one used for the system.

Being resistive sensors, the FSRs need to be supplied with a voltage. They also need to be attached to the subject and if we were to connect them to a fixed microprocessor we would need meters of cable. This would of course inconvenience the subject and affect the measurements. This means that to read the FS's signal we needed a portable power source and the capacity to send the recorded data to the controller responsible for the activation of the obstacle. Both these problems were solved by the OTB Sessantaquattro, a portable device for EMG recording with sixty-four acquisition channels and a sampling rate of 1 kHz. The Sessantaquattro is capable of wireless communication with a Personal Computer (PC), so it meant we had a way to record the signal and identify the time instants of interest in the gait. At the same time, this meant that we needed two separate control units since the Sessantaquattro could not be connected to the microprocessor reading the distance.

2.2.3 Actuators

The only instance in the project where actuation was needed was to operate the locking mechanism of the obstacle. Since the obstacles were spring-loaded, we needed a way to release them when we wanted a trip to happen. The locking mechanism was a metal pin hooked to the obstacle, which was under high strain because of the loaded state of the obstacle. To unlock it we decided to employ a solenoid valve in pull configuration, connected to the metal pin. The moving element of a solenoid valve is a ferromagnetic plunger inserted in a coil, which moves when a current runs through the coil. This kind of device was chosen because of its elevated pull speed and strength, which allowed a fast release of the obstacle. It should be noted though that the puling strength exponentially decreases the more the plunger is extracted from the coil. For this reason, great thought went into how much the plunger needed to be extracted when inserted into the hook and to minimize this amount. The force characteristic for the chosen solenoid valve is given in Figure 12.

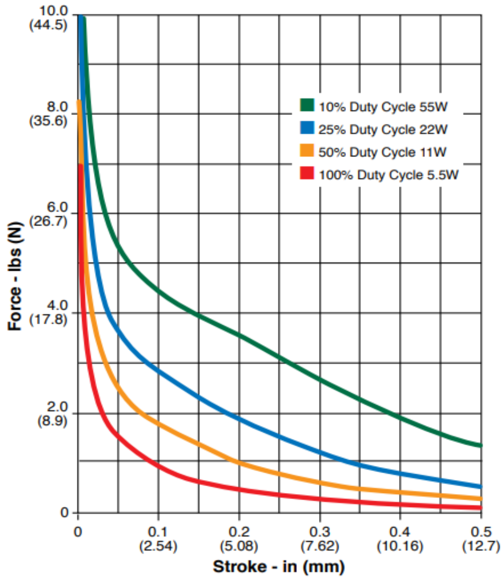


Figure 13, force profile of the solenoid valve.

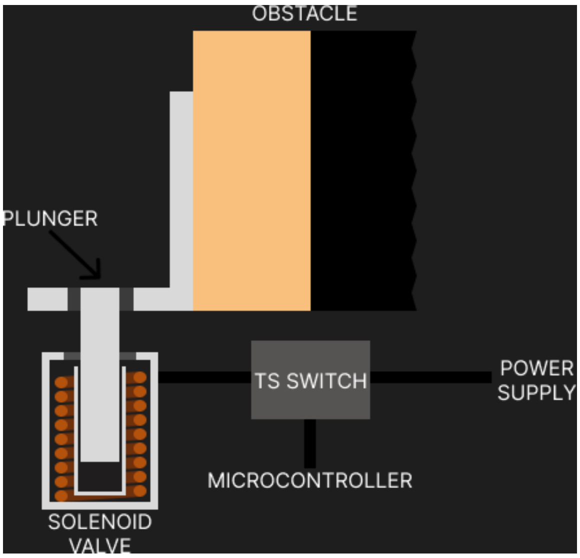


Figure 12, the schematic of the locking mechanism, with a focus on the electronics and the control system.

Since it needed to work intermittently, it was possible to operate it supplying with continuous current, which guaranteed maximum pulling force profile. The voltage necessary was 12 Volts, so we used a portable power supply to power up all the solenoid valves. Directly hooking the vales to the power supply would pull the plungers to their resting position, so transistor switches were used to control the activation of the solenoids. The switches were driven by a microcontroller, the same one that collected the distances from the distance sensors. A schematic for the locking mechanism is given in Figure 13.

2.2.4 IMUs

The final aim of our project was to obtain a setup that could serve as a base for the development of a tripping detection algorithm. While validation of the tripping setup could simply have been done through visual examination, we also needed to implement the data acquisition for the future part of the experiment. This way we would be able to check whether the data was being acquired properly and to start performing a first examination of said data, allowing us to know if the information extracted was useful. The data in question came from the IMUs we placed on the subject during the experiments. Depending on the trial, either three or four units were placed on the subjects, while one was placed on each obstacle to aid in the identification of the tripping instant.

The IMU system employed was the Xsens MTw Awinda. The IMUs of this system are equipped with a 3D rate gyroscope and a 3D accelerometer. In addition, they comprise a 3D magnetometer, a barometer, and a thermometer. The output of the measurement units, along with the calibrated magnetometer and barometer data, is transmitted wirelessly using the Awinda Protocol to the Awinda Master. The Awinda Master serves as the interface between the Awinda host (typically a PC running Xsens-based software, MT Manager in this case), and one or more IMUs. Through MT Manager it was possible to record the measurements from the IMUs. The Awinda Master ensures that the data from each IMU is synchronized to within 10 μs, guaranteeing an acquisition rate of 100 Hz. For this project, the Awinda Master used was the Awinda Station. It includes 4 BNC hardware connections that allow to send triggers to the Master. The triggers are then saved together with the samples. This way it was possible to synchronize the FS's signal and the IMU's signal, by sending a trigger to the Station when the obstacle was deployed.

2.2.5 Control units

From the sensors and the actuators' description, it is clear that two separate control units were needed to operate the system. The first control unit was a microcontroller, since both the distance sensor and the transistor switch needed to be hooked up to one. We decided to use an Arduino Uno, because of its simplicity and the fact that it was widely available in the laboratory where the project was developed. We also decided to assign every obstacle its microcontroller, for a series of reasons. First of all, a single microcontroller would not have had enough ports to connect all the sensors and the actuators. Furthermore, given the size of the whole setup, connecting all the peripherals to the same Arduino would have meant running meters of cables, which would have affected the traveling signal and the acquisition time for the sensors. Finally, the time between the readings performed on the distance sensors was 15 milliseconds. Because the distance signal needed to be filtered, the overall reading time ended up being 30 milliseconds (2.3.1). If we were to connect all the sensors to one Arduino, polling would have to be performed to read the distance of each sensor, introducing an overall delay in the code of over 100 milliseconds. This kind of delay was incompatible with our setup, as delaying the release of the obstacle could have meant missing the window to achieve impact with the foot.

The second control unit was a PC. The reason for this was that both the Sessantaquattro and the Awinda Stations needed a host to send their data to. The PC was connected to both of them and recorded their data, but it performed another task too. As stated above, the FS signal carried information on the gait phase. Since the Arduino could not connect to the Sessantaquattro, the analysis needed to be performed by the computer. For this reason, a Python script ran on it during the trial, tasked with identifying the gait phase for the tripped foot and sending this information to Arduino. So, while the Arduino decided whether the obstacle should be deployed, the PC sent the information necessary to make this decision and stored the data of interest produced by the experiment. A schematic of the flux of information for the control units is shown in Figure 14.

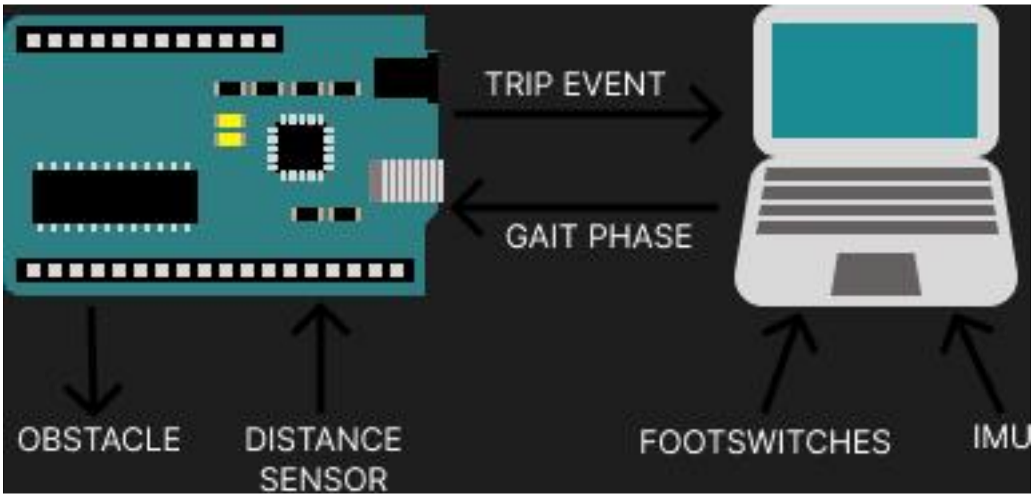


Figure 14, schematic of the interaction between the two control units.

2.3 Algorithms

The operation of the setup was managed by two algorithms, one for the PC and one for each control unit. The control unit algorithm running on Arduino was written in a variant of C++ and was responsible for deploying the obstacle. The one running on the PC was the Python script introduced in the previous section. Both of them went to different iterations as the system evolved, particularly the Python script.

2.3.1 Arduino script

To deploy the obstacle, the algorithm needed to close the digital switch (i.e., a transistor) connecting the power supply and the solenoid. This had to be done only when there were the right conditions for causing a trip. So during each iteration, the algorithm checked whether the conditions were verified and released the obstacle at the right moment.

An Arduino script is usually divided into a setup section and a looping section. Before the setup though, the necessary libraries have to be loaded, and the variables and constants that the code needs have to be defined. Then, in the setup section, the necessary ports are defined as inputs or outputs, and the communication lines are opened. For this project, the distance sensor communicated with the Arduino through the I2C protocol, so the Wire library was used. The Arduino and the PC also needed to communicate between themselves: on one side, the Arduino needed to know the gait phase, while the PC needed to know when the tripping device activated to save the data and interrupt the execution of its script. This communication used the UART protocol instead.

At the start of the looping section, the algorithm checked the content of the serial line. If there was something on it, it meant that the PC was trying to send information to the Arduino. More specifically, three pieces of information could be shared by the PC: the occurrence of a heel strike event, the occurrence of a toe-off event, or the acknowledgment that the data sent by the Arduino had been stored. The first two pieces of information effectively changed the status of the system: the time at the instant of arrival of the information was saved, either as the toe-off time or the heel strike time. Whichever of the two was greater signaled whether the tripped leg was in the swing phase or the stance phase.

Algorithm 1 Threshold updater

If *message available* **then**
    If *toe-off* **then**
        Update toe-off instant (TOI)
    Else if *heel strike* **then**
        Update heel strike instant (HSI)
    Else if *acknowledgment* **then**
        Send trigger to IMU station
    End if
    Compute swing time as HSI-TOI
    If *swing time > 0* **then**
        Store current swing duration with the swing durations of the last nine steps
        Compute median swing time from the last ten recorded swing times
        Compute swing threshold as 60% of median swing time
    End if
End if

As stated before, only during the swing phase a trip could happen. Furthermore, most of the studies on the topic tried to elicit a trip only in the early- and mid-swing phases. We decided to follow this trend, meaning we had to check how long the swing phase had lasted before activating the device. Mid-swing phase ends at approximately sixty percent of the overall swing phase, so this value would constitute the threshold for the activation of the device. Since the swing duration is not a constant value, the threshold had to be adaptive and updated during the trials. Instead of a fixed swing duration, we calculated the median swing duration, calculated from ten consecutive steps.

The actual threshold was sixty percent of this value. The update was done each time a swing ended, so whenever Arduino received the signal of heel strike. This portion of the algorithm is schematically described in Algorithm 1

The second part of the looping section dealt with the activation of the device. First, the distance was retrieved from the distance sensors. As stated in previous sections (2.2.2), filtering needed to be performed on the distance data. The purpose of it was to eliminate the outliers since they could result in a faulty activation. The filter applied was a median filter taking as input 3 consecutive distance readings. This number of sample was selected to minimize the delay since each reading of the sensor took 15 milliseconds. After filtering, if the distance obtained was under 20 centimeters, the algorithm proceeded to the next check for the release of the obstacle.

Having confirmed that the tripping foot was approaching the obstacle, the algorithm had to check that it was in the right phase of the swing. So at the start of each iteration, whenever the tripping foot was in the swing phase, the system computed how long the swing phase had lasted. If the distance recorded was below 20 centimeters and the foot was in early or mid-swing, the obstacle could be deployed. To further reduce anticipatory movements we implemented a randomization factor, which prevented the device from activating every time the first two checks were cleared. This was introduced via an additional check on a randomly generated variable.

If all three checks were satisfied, the algorithm deployed the obstacle, by closing the transistor switch. At the same time, a message was sent to the PC, to inform it of the activation of the obstacle.

Algorithm 2 Obstacle deployer

```
Read current distance value and store it with last two measurements
Compute the median distance
If median distance < 20cm then
    Compute current swing duration
    If current swing duration < swing threshold then
        Generate random value
        If random value < frequency threshold then
            Activate the obstacle and signal activation to the PC
        End if
    End if
    Send current swing duration, swing threshold and random value to PC
End if
```

A message was also sent by the Arduino each time the distance recorded went below the threshold level. This message contained the duration of the swing when passing in front of the sensors, the threshold for the mid-swing and the random value used for activation. This information served two purposes: to check the proper operation of the system during the trials and to send a trigger to the Awinda Station. As previously described, a trigger was sent by the Arduino whenever an acknowledgment for the message was received. This second section of the Arduino algorithm is described in Algorithm 2 .

2.3.2 Python script

The Python script's main tasks were retrieving the signal from the FS, performing analysis on it to identify heel strike and toe-off events, and saving the experimental data. Since these operations needed to be performed at the same time, we decided to subdivide the algorithm into three subprocesses: the acquisition process, the control process, and the saving process.

The acquisition process and the saving process were fairly straightforward. When the connection with the Sessantaquattro was established, the acquisition process recovered the data from it and stored it in two queues. These queues made the data available to the other two subprocesses. One was made for each since retrieving the data meant removing it from the queue. The saving process was instead responsible for saving the raw data coming from the Sessantaquattro and the information sent by the Arduinos. Every piece of information was saved together with the saving time, to facilitate the analysis of the data.

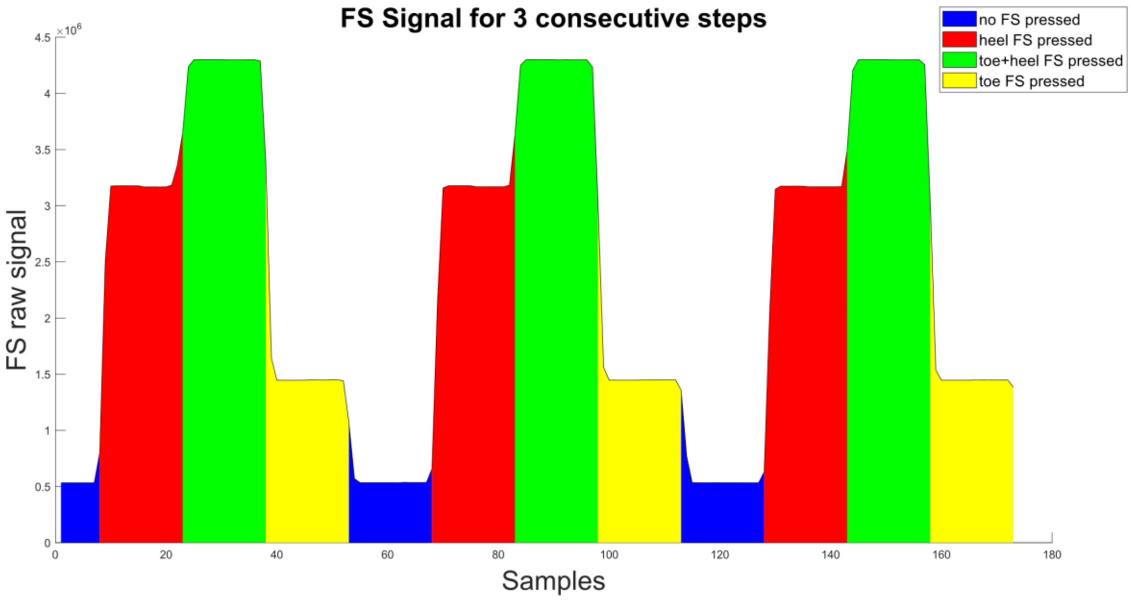


Figure 15, the FS signal for 3 consecutive steps. Each color zone represents a different combination of the activation of the FS. The blue zone corresponds to the swing phase, while the other correspond to the stance phase.

The control process was the most substantial section of the Python algorithm. It dealt both with the analysis of the FS signal and the communication with all the Arduinos in the system. Firstly, a connection with the Arduinos was established. Then the algorithm entered in a loop that remained active until the activation of the obstacle. In each loop, the data was extracted from the data queue. Ideally, the signal from the FS should have been constituted by a sequence of square waves: the various combinations of activation of the two sensors resulted in square waves of different amplitude, as shown in Figure 15. This meant that we could have used the derivative of the signal to identify the toe-off and the heel strike. Unfortunately, FSRs are not extremely reliable, meaning the rising and falling front of the square waves were not particularly sharp. If we were to use the derivative of the raw signal, it would have been difficult to identify a reliable criterion for identifying the relevant gait event. For this reason, we decided to perform quantization on the signal. The effect of quantization is shown in Figure 16.

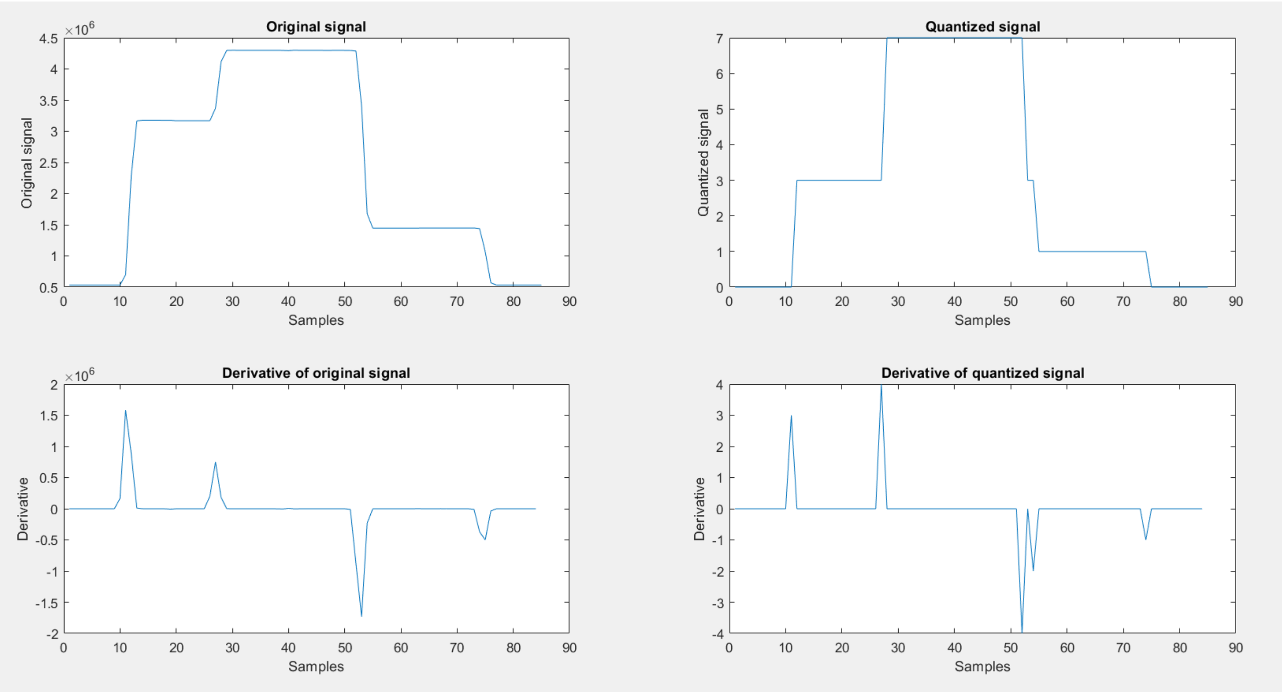


Figure 16, the FS signal before (left) and after quantization (right). On the bottom, it is possible to see the effect of the quantization on the derivative.

After quantization, the derivative of the signal was analyzed. Depending on the value assumed, we were capable of identifying a toe-off or a heel-strike event. Since the queue contained only 20 milliseconds, only a single toe-off or heel strike should have been identified in this window. If more were detected, only the first was considered, as the others represented an error in the signal. After detecting an event, a message was sent to all the Arduinos, containing either the character “1” or the character “0”. Character “0” represented a heel strike, while character “1” represented a toe-off.

After sending the message, the algorithm checked whether any of the Arduinos were trying to communicate with the PC. If one of the Arduinos was detected trying to send a message, the message was read and stored in a queue shared with the saving process. After receiving the message, an acknowledgment is sent back to the Arduinos. If the message informed of the deployment of one of the obstacles, the control loop was interrupted, after storing the last bit of data contained in the message. The totality of the algorithm is described by Algorithm 3.

All the subprocesses were initialized in the main of the script, which was also responsible for opening the communication channel with the Sessantaquattro and constituted the intermediary between the operator and the algorithm. Through the main, the operator could turn on the whole system, set the name for the saving file, and interrupt the trial if necessary.

Algorithm 3 Control subprocess

```
Establish connection with the Arduinos
While no obstacle was activated do
    Read the FS signal from the queue and quantize it
    Compute the derivative of the signal and eliminate outliers
    If derivative == heel strike derivative then
        Send a "0" to the Arduinos
    Else if derivative == toe-off derivative then
        Send a "1" to the Arduinos
    End if
    If message from Arduino was received then
        Read the message
        If message == "A" then
            An obstacle was activated
        End if
        Store the message in a save queue and send acknowledgment to Arduino
    End if
End while
```

2.4 Experimental protocol

While the main objective of this project was the validation of the tripping setup, we also had to test the procedure for collecting the data for the trip detection algorithm. Furthermore, the preliminary data collected in this phase could be used to preemptively determine if the setup would produce meaningful data. The following experimental protocol was developed with these objectives in mind.

2.4.1 The protocol

The procedure followed for every subject is described below:

1. The subject was introduced to the experimental setup and the experimental protocol was explained to them.
2. The subject was asked to sit and the FS were fixed under the tripping foot with duct tape. One of the FS was placed under the heel and the other under the toe. We decided that for healthy subjects the tripping leg was always going to be the right leg. For future tests performed on amputees, the tripping leg will be the prosthetic one.
3. Elastic bands were placed on the tripping leg and around the waist. These served to secure the cables connecting the FS to the Sessantaquattro, to allow for placement of the IMU and of the Sessantaquatrro itself.
4. The Sessantaquattro was attached to the waistband and the four IMUs were placed on the subject. The placements for the IMUs were: dorsum of the foot; laterally on the shank, just below the knee; laterally on the thigh, at the same height as the groin; on the back of the subject, in correspondence with the L-2 vertebra. The placement is shown in Figure 17.
5. To check the correct placement of the FS, the patient was asked to walk on the walkway. This way we could check the signal before proceeding with the trials. Furthermore, this allowed the subject to familiarize with the setup.
6. At the beginning of each trial, we had the patient stand still on one side of the walkway. From that moment, we started the video and the IMU recordings. After this, we launched the Python script that controlled the system. After hearing an acoustic signal generated by the script, the subject could start walking on the walkway.
7. During each trial, the subject had to walk on the walkway in circuits. They had to keep walking until either the system activated or the operator instructed them to stop. The subject was instructed to keep walking for a couple of seconds after the tripping event. This was done to mimic the behavior after a real trip and acquire data on the response to the trip. The first trial was always a mockup trial without activation, to collect baseline data on the gait. To keep the subject attention away from the obstacles, we distracted them through conversations. This kept them distracted without significantly affecting their gait.
8. When the subject stopped walking, the trial ended. Both the recordings and the script were interrupted, and the data was stored. A questionnaire was given to the subject after each successful trippin trial.
9. Steps 6 to 8 were repeated until five successful tripping trials were recorded. Afterward, the sensors were removed from the subject and the experiment was interrupted.

Regarding the walkway circuit that the subjects were asked to walk along, some further specifications are needed. Two lines were drawn on the walkway, both running parallel to the longer sides of the platform. Each line was drawn 20 centimeters from the side and the subject was asked to walk keeping their tripping foot in the zone between the line and the side of the walkway. The supported foot instead had to stay in the zone between the two lines (i.e. the middle of the walkway) at all times. The subject had to walk from one side of the platform to the other and back, keeping their feet in the assigned zones. This way we made sure the tripping foot was constantly both in range of the obstacle and the distance sensor.

2.4.2 The questionnaire

At the end of each trial, we had the subject fill out a form. The form consisted of four questions:

1. On a scale from 1 to 5, how much were you surprised by the obstacle activation?
2. On a scale from 1 to 5, how close to a real-life stumble was your impact with the obstacle?
3. On a scale from 1 to 5, do you feel that your reaction to the stumble was natural?
4. On a scale from 1 to 5, did you feel any discomfort upon impacting the obstacle?

The purpose of the questionnaire was twofold: it allowed us to gauge the level of adaptation of the subject for consecutive trials and to evaluate the realism of the stumble from the subject point of view. The last question was included to be sure that the impact with the obstacle was not painful for the subject.

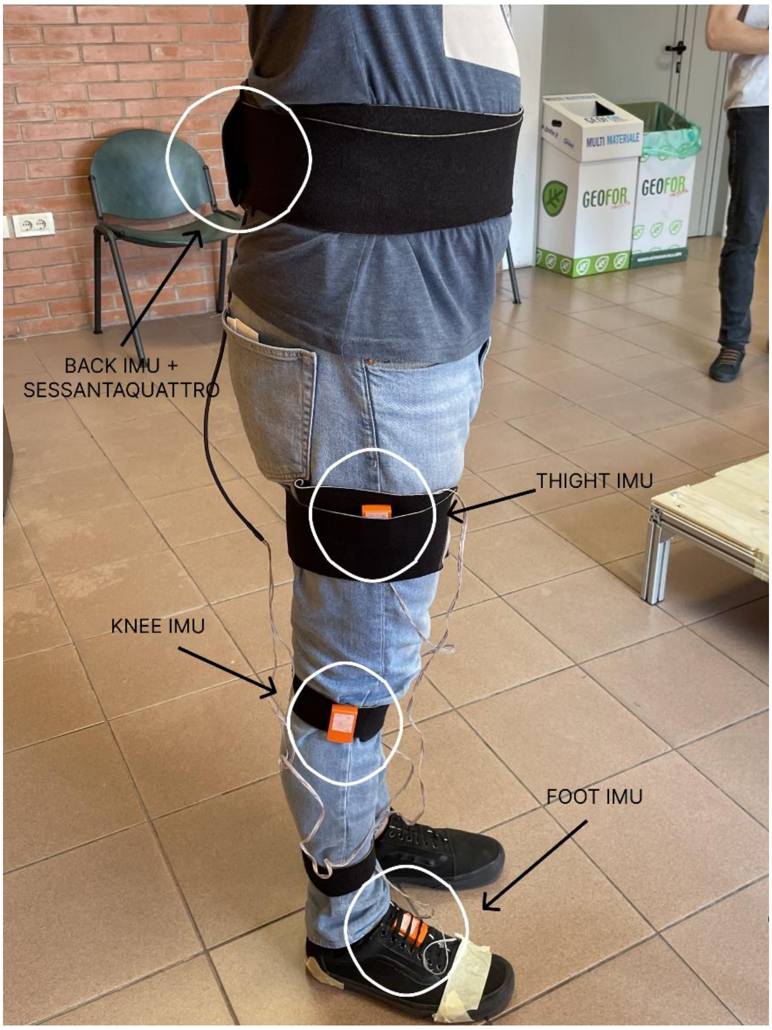


Figure 17, placement of the four IMUs and the Sessantaquattro on one of the subjects.

2.5 Analysis

During the experimental session, data from eleven subjects have been acquired. A total of eighty trials were recorded and sixty answers from the questionnaire were collected. The mismatch came from the fact that only successful tripping events were followed by filling up the form: faulty activations and mockup trials had no form associated with them.

2.5.1 Data

The output of each trial was a track from the FS and a track for each IMU involved in the trial. As stated previously an IMU was fixed to each of the obstacles, but for some trials one obstacle had to be removed due to damage to the rotating arm. Furthermore, while technically four IMUs were to be placed on the patient, only seven were available for half of the experiments. Excluding the four placed on obstacles, only three remained to be placed on the subject. Since we were mostly interested in data from the leg, we didn't place an IMU on the back when not possible. This meant that for each trial we had between seven and eight tracks from the IMUs. Each IMU track was a bidimensional array, as each IMU measured the accelerations and the angular velocities along the three Cartesian axes, for a total of six signals.

2.5.2 Processing

To process the data from the experiment, two MATLAB scripts were developed. The first synchronized the FS and IMU signals and identified the instant of the trip. We were interested in extracting all the steps from each trial from the IMU signal and to do so we intended to use the instants of toe-off and heel strike. These were contained in the FS signal, meaning we could use them only if the IMU and FS track had the same size and sample frequency. A schematic description of how the first algorithm operated can be found in Algorithm 4.

Algorithm 4 Synchronizer

1: Load the FS and the IMU data from one trial

2: Identify the FS sample corresponding to obstacle activation

3: Identify the IMU sample corresponding to the last trigger

4: Resample the IMU signal and resize the FS and the IMU array to match the previously obtained samples.

5: **If** *not first trial* **and** *obstacle caused a trip* **then**

6: Identify the IMU associated with the deployed obstacle

7: Compare the z-acceleration of the IMU with the mean z-acceleration when no impact occurred

8: Identify the sample corresponding to the trip

9: **End if**

2.5.2.1 Synchronization

To perform the alignment, the algorithm operated in two steps. For both the IMU and the FS signals the samples associated with the activation of the devices were recorded during the trial. For the IMU signal, the sample of activation was obtained thanks to the trigger sent to the Awinda Station. For the FS, the time of the activation was saved and compared with the time each sample was recorded. To align the two signals, they were cut to have the samples corresponding to the tripping event coincide. Before doing this though, interpolation had to be performed on the IMU signal, as its sampling time was 100 Hz and the one of the FS was 1 KHz. This technique was quite unreliable though, because of the delay between the activation of the obstacle and the time necessary to send the information to the devices that recorded it.

This first alignment was not extremely accurate, and a certain degree of mismatch remained between the two signals, mostly due to the delay between the activation of the obstacle and the recording of the activation. To eliminate the mismatch, the angle of the foot was computed from the angular velocity of the IMU placed on the foot. According to [29] the toe-off corresponds to the moment where the foot reaches the maximum angle with respect to the transverse plane. Since each trial began with the subject standing still it was possible to identify the sample of the first toe-off in both the angle signal and the FS signal. By matching these samples it was possible to eliminate the residual mismatch from the first aligning process. The first algorithm ended its task by saving the synchronized and resampled FS and IMU arrays.

2.5.2.2 Trip recognition

To identify the tripping instant we developed a trip detection algorithm that made use of the IMUs placed on the obstacles. Before performing the experiments, we recorded the acceleration signal corresponding to the release of the obstacle without an impact with the foot. We recorded 10 tracks and we extracted the mean and the standard deviation of the signal from them. The reasoning behind this was that the activation with the impact would have a similar acceleration profile to a release without impact until the instant of contact with the foot. It was then possible to identify the exact instant of the trip as the instant when the signal from the trial diverges by the mean signal by a value greater than the standard deviation. The comparison between the two signals is shown in Figure 18. With this second technique, we could associate the exact instant of the trip with one of the IMU samples. After this, synchronization was performed using this sample instead of the last trigger sample. This technique was preferable over the first as it returned the tripping instant and it had a greater reliability compared to the first. Unfortunately, it had two limitations: first, a trip needed to happen, which was not the case for the mockup trials; second, if the IMUs on the activated obstacle did not record properly it was impossible to perform this technique.

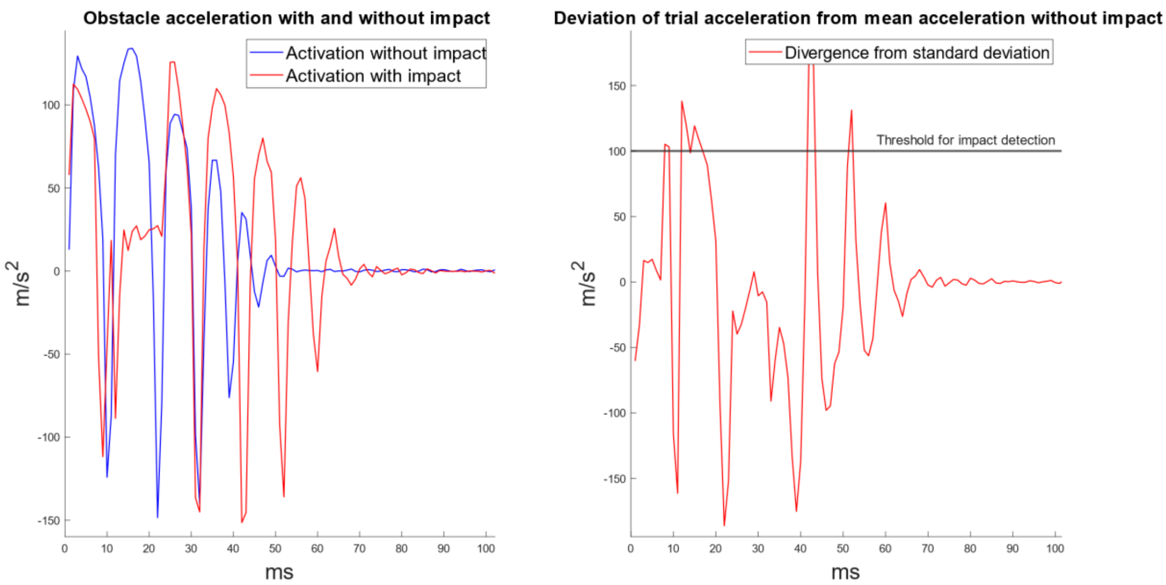


Figure 18, tangential acceleration of the obstacle during deployment. On the left the red signal represents the acceleration of the obstacle during a trial, while the blue signal represent the average acceleration for the obstacle when deployment happened without causing a trip. On the right the red line represents the difference between the trial acceleration and the average acceleration minus the standard deviation.

2.5.2.3 Step extraction

The second algorithm takes these arrays as inputs. Its purpose was to reorganize the IMUs in a comprehensive array for each subject, where each step could be individually selected. The algorithm started by identifying the indexes of samples corresponding to all heel strikes and toe-offs from the FS signals. Thanks to the alignment, these could used to isolate the single steps from the IMU tracks. The isolated steps were then resampled to obtain signals with the same number of samples. The resampled signals were 2000 samples long, with the first 800 samples corresponding to the swing phase and the last 1200 to the stance phase. These numbers were selected to maintain the proportions for the phase durations that are typically found in a gait cycle. Each step track was actually 2005 samples long, as 5 more values were added to the end of the array. This tail contained information on the step: the trial it was recorded in, the original swing and stance lengths, if it preceded or followed the tripping step, and the tripping instant. For each subject, the resampled steps from all trials were collected in a single tridimensional array.

2.6 Results

Two pieces of data were obtained from processing: the arrays generated by the processing of the recorded data and the results of the questionnaire. In total, we collected sixty forms and recorded more than two thousand steps from fifty-seven separate trials. All data collected is summarized in Table 1.

Table 1, data collected from the experiments.

Subject	Trials recorded	Trials after processing	Steps in final array	Forms filled
1	9	7	231	9
2	8	7	171	6
3	6	4	113	4
4	6	6	183	5
5	7	4	106	6
6	10	0	0	6
7	7	6	558	5
8	6	5	337	3
9	7	6	0	4
10	8	6	236	6
11	6	6	328	6
Total:	80	57	2236	60

2.6.1 IMU signals

Of the original eleven subjects, only nine remained after the processing step, and forty-two tripping steps were collected. Subject 6 was excluded before the processing stage, due to equipment malfunction during the experiment; subject 9 was discarded after processing because the data associated with them presented an abnormal distribution.

Over 50% of the recorded trips happened in the early swing and 40% happened mid-swing. Only two trips took place in the late swing. Given that the objective of the project was the validation of the system, the analysis of the steps was mainly carried out by visual investigation of the IMU signals. We decided to compare the tripped steps with the standard steps, starting with a single subject. We computed the mean and the standard deviation for all the IMU's signals of trip-less steps, excluding the steps following a trip in each trial. The two values combined were plotted, together with the signals corresponding to some of the tripped steps. The objective was to identify significant differences between trip and trip-less signals. An example of the output of this process is shown in Figure 19. The blue line and the shaded area represent the combination of the mean and the standard deviation of all trip-less steps, while the other lines correspond to three different trips. The quantities displayed were chosen for the sake of this work, since it would have been unfeasible to show all the plots in a single image.

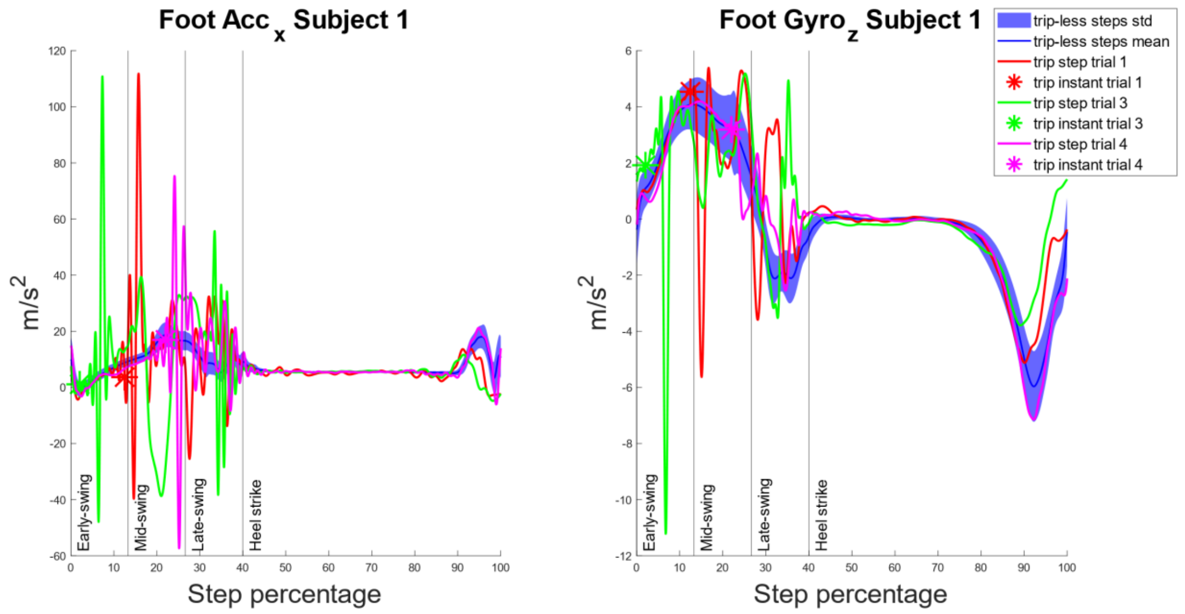


Figure 19, x-acceleration and y-angular-velocity of the foot during trips versus trip-less steps for one subject. The instant the trips happened are signaled by stars on the graphs.

Since the detector had to work regardless of the subject, we were interested in knowing how tripped steps from different subjects compared to regular steps of all subjects. Therefore we repeated the same analysis grouping the subjects together. This gave us a more general profile for a trip-less step, performing mean and standard deviation from most steps of the subject, again excluding post-trip steps. A sample of this analysis is displayed in Figure 20. The quantities selected for display are the same as the single subject, for coherency's sake.

From this analysis, it emerged that the accelerations and angular velocities' mean profiles were similar across all subjects when no trip happened. By comparing the standard deviation for a single subject and for all the subjects it was possible to say that on average the inter-subject variability was greater than the intra-subject variability.

For what concerned the signals associated with the trips, their profiles were not comparable. The reason for this is that each trip happened in a different portion of the step and that subjects had different weights and heights, which affected the dynamic of their limbs. Some general information can still be retrieved though. Acceleration and angular velocities signals had a higher frequency content when the subject tripped on the obstacle, as this introduced rapid changes in the trajectory of the bodily segments. For this reason, the signals for most of the tripped steps presented sharp peaks and fast oscillations. While inter-trip comparison was not feasible, it was still possible to compare signals recorded by different IMUs for the same trip. As we can see in Figure 21, perturbations appear as a series of oscillations in the acceleration profiles of the shank and the thigh. If we consider the delay between the instant of the trip and the onset of the perturbation, it seems that this increases as we get further away from the point of impact, i.e. the foot. At the same time, the magnitude of the perturbations decreases.

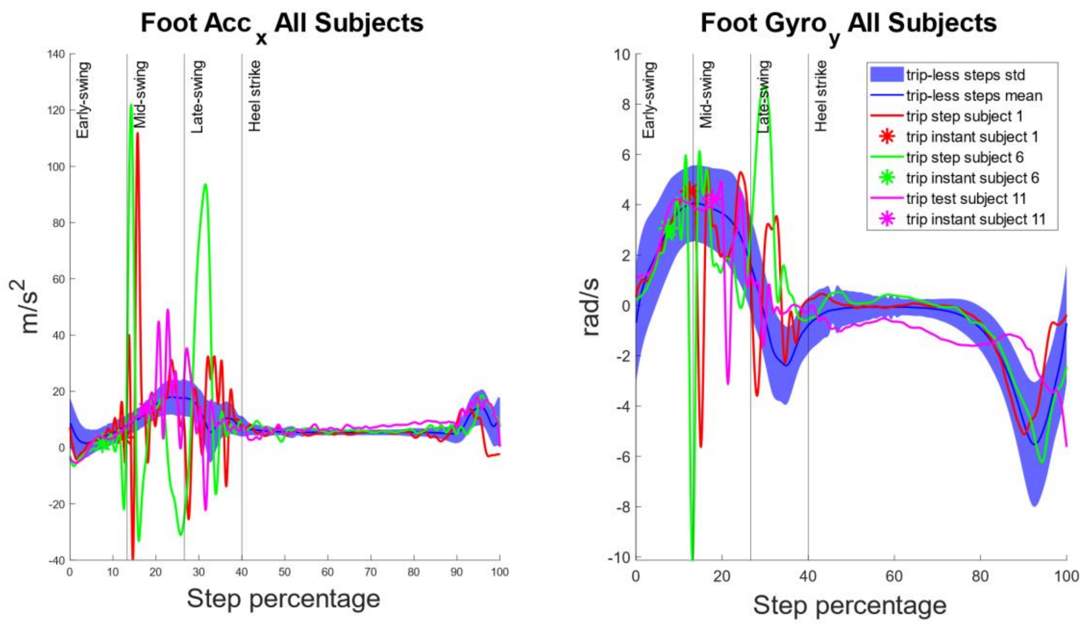


Figure 20, x-acceleration and y-angular-velocity of the foot during trips versus trip-less steps for all subjects. The instant the trips happened are signaled by stars on the graphs.

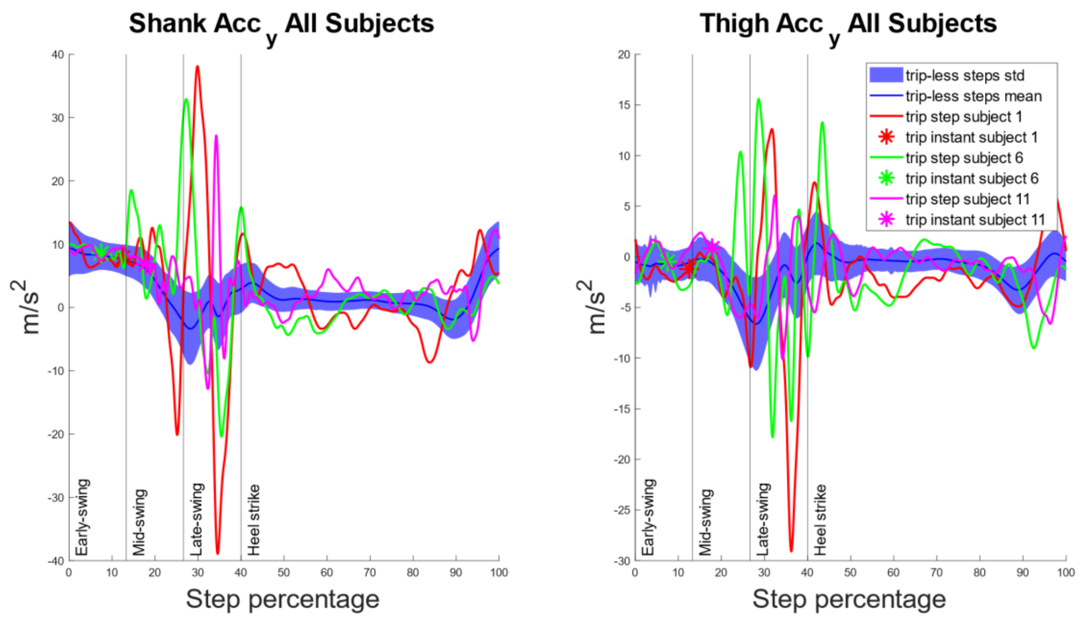


Figure 21, y-acceleration of the shank and the thigh during trips versus trip-less steps for all subjects. The instant the trips happened are signaled by stars on the graphs.

2.6.2 Questionnaire results

On the questionnaire side, we were interested in two things: how the level of surprise evolved across consecutive trials and how the subject perceived the impact with the obstacle. The subjects expressed their level of surprise and the realism of the impact through a scale ranging from one to five. The level of surprise was investigated by plotting the median level of surprise for each trial, considering the answers given by all the subjects. If we consider the Inter Quartile Ranges of the answers for each trial, visible in Figure 22, the first quartile is at least equal to three for all trials, excluding the trials where only one subject answered the questionnaire. In general, the level of surprise seems to decrease if we consider the combination of the IQR and the median, although the trend is fairly slow. To investigate the realism of the impact, we created a pie chart, presented in Figure 23. As we can see, no subject attributed the minimum value of realism to the trip, and more than 60% of the trips were considered to have a level of realism of at least four.

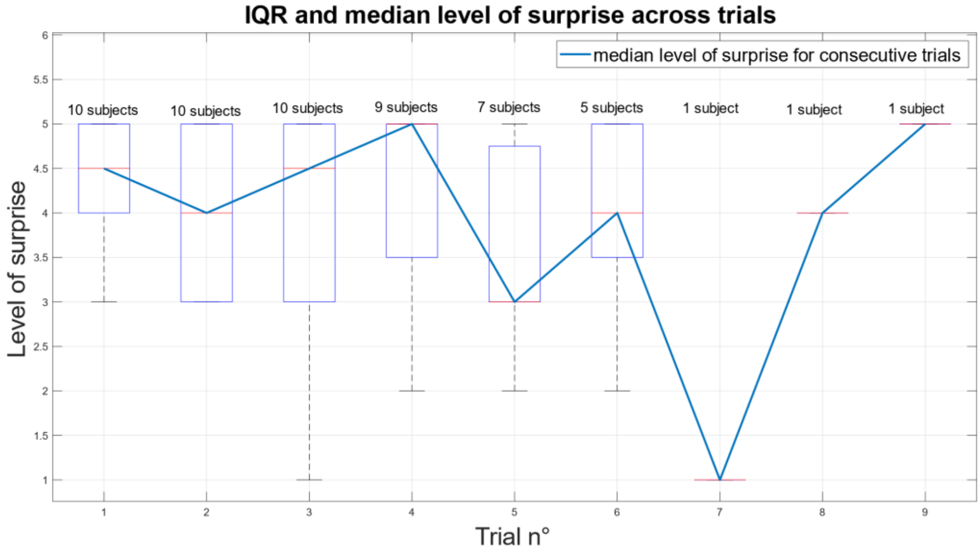


Figure 22, Inter Quartiles Ranges (IQR) and median of the level of surprise to the tripping event reported by the subjects for each trial.

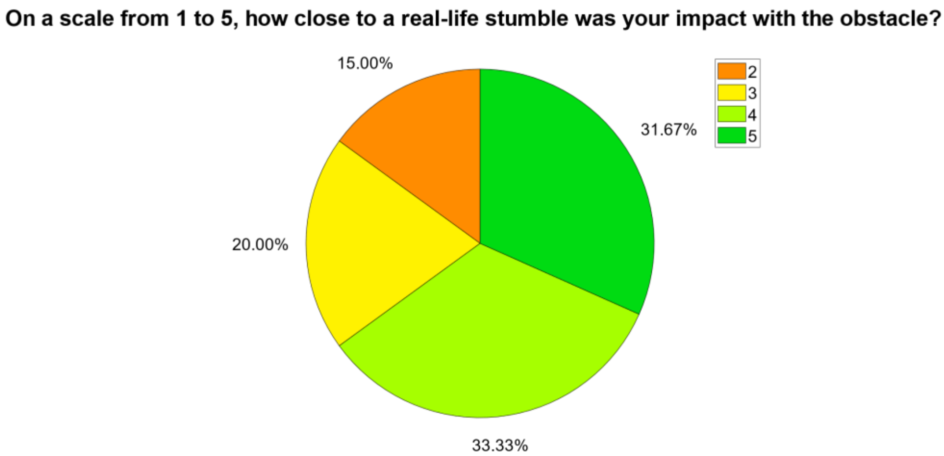


Figure 23, perceived realism of the trip.

3. Conclusions and future work

The objective of this project was the development of an experimental setup capable of consistently causing realistic trips. By simply observing the trials, it was already possible to say that our system caused stumbles, which meant that we achieved our objective. Of course, consistency and realism had to be investigated too, and the questionnaire came to our aid for this. Overall, the subjects were considerably caught by surprise by the activation of the obstacle even after numerous trials. The level of surprise decreased as trials went on, but the decrease was slow enough to say with confidence that our system did not lose effectiveness as the experiment progressed. The subjects seemed to consider the trip fairly realistic, as most of the trips were attributed an above-average realism with respect to the range of our scale.

Overall, this meant that we were successful in simulating realistic trips, but the final application of this setup had to be considered too. While we could have stopped the project strictly at the validation phase, we were interested in knowing if the setup developed could be useful in obtaining data for the tripping detection algorithm. This was the aim of the visual analysis performed on the quantities measured by the IMU. As we can see in Figure 19, Figure 20, and Figure 21, the signal associated with the trip deviates significantly from the average step profile. The deviation is significant enough to let us believe that some feature could be extracted for the detection of the trip, although no specific analysis was made in that sense. For this reason, we can say with sufficient confidence that the setup constitutes a solid foundation for future work on the parent project.

Some educated guesses can also be made from the visual analysis. First of all, the magnitude of the perturbation with respect to the average signal is much greater for the accelerations than for the angular velocities. This suggests that accelerometers could be more useful to identify trips, which is advantageous given the higher cost of gyroscopes. Also, the magnitude decreases the further the sensor is from the point of application of the perturbation. Combining this information with the fact that each body segment introduces a delay in the onset of the perturbation, it seems reasonable to assume that the sensor should concentrate around the foot. This way the trip will be easier to detect, both in terms of signal discrimination and detection speed. Speed will likely be a relevant factor in the development of the detector, as there is a limited amount of time to react to the perturbation.

While the results obtained are promising, the setup could still be perfected. Particularly, an issue is still present: faulty activations. While walking on the walkway, the subjects produced vibrations. These unexpectedly caused the uncoupling of the locking pin and the obstacle hook, since their contact surface had to be kept to a minimum. This problem could be solved by employing stronger solenoids: this would allow us to have a sturdier coupling between the locking elements, without compromising the controlled release of the obstacle.

Other than this problem though, the setup is ready to be employed in the collection of the actual data that will be used for the development of the algorithm. Not only that, the work made to process the data collected for this project constitutes a framework for future data analysis and the visual analysis provided significant insight that could be useful for the development of the detector. While this project was a preliminary step of the parent project, I believe the work done could have a significant impact on the future development of LLPs. The development of trip-reacting prostheses could increase the faith of amputees in active prostheses, in a world where improvement in mobility is often sacrificed in favor of simplicity. Not only that, this setup could be replicated to reliably study trips for a variety of subjects, as increasing our knowledge in the field could aid in reducing fall risk.

Acknowledgments

I would like to express my sincere gratitude to my advisors, professors Alessandra Laura Giulia Pedrocchi and Silvestro Micera. They provided me with the invaluable opportunity of working on an innovative project in one of the most prestigious institution in the country, and for that I will be forever grateful. This project would have not been possible without their guidance and support.

I would like to extend my thanks to Eugenio Anselmino, who followed me throughout every step of the project. His help was key to the development of the whole experiment and thanks to him I was capable of greatly developing my own skills. My thanks also go to the whole Sant'Anna Institute, for the help they gave me with the experiment and for making me feel like a member of the lab during my permanence.

Next, my thanks go to my family: my parents, Antonio and Cristina, and my sibling Nicole. The love and support they always provided me with were fundamental to the completion of my university career. They granted me the opportunity to pursue my dreams and aspirations and were always there for me for everything I ever needed. I am grateful to the rest of my family as well, particularly to my grandfather Bruno and my aunt Grazia for always believing in me.

Then I would like to thank my friends. To Maria, my best and closest friend. Had I not met you I would not be the person I am today: you made my life better in any way imaginable and I am extremely happy to call you my friend. I know I can always count on you, for a laugh, a confidant, or a cry. To Beatrice, an unexpected and invaluable friend. I cannot imagine my life without you and I can always count on you if I need a fun night out. You are the best party member anyone could hope to have. To Marta, my crafting buddy. The last year would have not been the same without you, I was really happy to have someone to accompany me in my creative endeavors. And to all my friends, for making life on this planet much more enjoyable.

Bibliography

[1] J. Kim, M. J. Major, B. Hafner, and A. Sawers, "Frequency and Circumstances of Falls Reported by Ambulatory Unilateral Lower Limb Prosthesis Users: A Secondary Analysis," *PM and R*, vol. 11, no. 4, pp. 344–353, Apr. 2019, doi: 10.1016/j.pmrj.2018.08.385.

[2] M. McGrath *et al.*, "Can microprocessor knees reduce the disparity in trips and falls risks between above and below knee prosthesis users?," *PLoS One*, vol. 17, no. 9 September, Sep. 2022, doi: 10.1371/journal.pone.0271315.

[3] W. C. Miller, M. Speechley, and B. Deathe, "The prevalence and risk factors of falling and fear of falling among lower extremity amputees," *Arch Phys Med Rehabil*, vol. 82, no. 8, pp. 1031–1037, 2001, doi: 10.1053/apmr.2001.24295.

[4] A. Gaetano, "Relationship between physical inactivity and effects on individual health status," *Journal of Physical Education and Sport*, vol. 16, pp. 1069–1074, Oct. 2016, doi: 10.7752/jpes.2016.s2170.

[5] F. Im Breisgau, J. Ee--, N. Thatte, N. Srinivasan, and H. Geyer, "Robotics: Science and Systems ssss Real-time Reactive Trip Avoidance for Powered Transfemoral Prostheses," 2019. doi: 10.0/Linux-x86_64.

[6] "Minimum foot clearance during walking, strategies for the minimisation of trip related falls".

[7] A. M. Schillings, B. M. H. Van Wezel, T. H. Mulder, and J. Duysens, "Muscular Responses and Movement Strategies During Stumbling Over Obstacles," 2000.

[8] C. Shirota, A. M. Simon, and T. A. Kuiken, "Trip recovery strategies following perturbations of variable duration," *J Biomech*, vol. 47, no. 11, pp. 2679–2684, Aug. 2014, doi: 10.1016/j.jbiomech.2014.05.009.

[9] C. Shirota, A. M. Simon, and T. A. Kuiken, "Transfemoral amputee recovery strategies following trips to their sound and prosthesis sides throughout swing phase," *J Neuroeng Rehabil*, vol. 12, no. 1, pp. 1–11, Dec. 2015, doi: 10.1186/s12984-015-0067-8.

[10] B. Balaban and F. Tok, "Gait Disturbances in Patients With Stroke," *PM and R*, vol. 6, no. 7. Elsevier Inc., pp. 635–642, 2014. doi: 10.1016/j.pmrj.2013.12.017.

[11] P. H. Sessoms *et al.*, "Method for evoking a trip-like response using a treadmill-based perturbation during locomotion," *J Biomech*, vol. 47, no. 1, pp. 277–280, Jan. 2014, doi: 10.1016/j.jbiomech.2013.10.035.

[12] S. L. Arena, J. L. Davis, J. W. Grant, and M. L. Madigan, "Tripping elicits earlier and larger deviations in linear head acceleration compared to slipping," *PLoS One*, vol. 11, no. 11, Nov. 2016, doi: 10.1371/journal.pone.0165670.

[13] S. L. Arena, J. L. Davis, J. W. Grant, and M. L. Madigan, "Tripping elicits earlier and larger deviations in linear head acceleration compared to slipping," *PLoS One*, vol. 11, no. 11, Nov. 2016, doi: 10.1371/journal.pone.0165670.

[14] B. E. Lawson, H. A. Varol, F. Sup, and M. Goldfarb, "Stumble detection and classification for an intelligent transfemoral prosthesis," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC'10*, 2010, pp. 511–514. doi: 10.1109/IEMBS.2010.5626021.

[15] O. Aziz, E. J. Park, G. Mori, and S. N. Robinovitch, "Distinguishing the causes of falls in humans using an array of wearable tri-axial accelerometers," *Gait Posture*, vol. 39, no. 1, pp. 506–512, Jan. 2014, doi: 10.1016/j.gaitpost.2013.08.034.

[16] J. K. Lee, S. N. Robinovitch, and E. J. Park, "Inertial Sensing-Based Pre-Impact Detection of Falls Involving Near-Fall Scenarios," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 2, pp. 258–266, 2015, doi: 10.1109/TNSRE.2014.2357806.

[17] A. F. Cordero, H. F. J. M. Koopman, and F. C. T. Van Der Helm, "Multiple-step strategies to recover from stumbling perturbations." [Online]. Available: www.elsevier.com/locate/gaitpost

[18] T. M. Owings, M. J. Pavol, and M. D. Grabiner, "Mechanisms of failed recovery following postural perturbations on a motorized treadmill mimic those associated with an actual forward trip." [Online]. Available: www.elsevier.com/locate/clinbiomech

[19] B. C. Lee, B. J. Martin, T. A. Thrasher, and C. S. Layne, "The effect of vibrotactile cuing on recovery strategies from a treadmill-induced trip," *IEEE Transactions on Neural*

Systems and Rehabilitation Engineering, vol. 25, no. 3, pp. 235–243, Mar. 2017, doi: 10.1109/TNSRE.2016.2556690.

[20] F. Zhang, S. E. D'andrea, M. J. Nunnery, S. M. Kay, and H. Huang, "Towards design of a stumble detection system for artificial legs," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 5, pp. 567–577, Oct. 2011, doi: 10.1109/TNSRE.2011.2161888.

[21] "Minimum toe clearance adaptations to floor surface irregularity and gait speed".

[22] B. W. Schulz, "A new measure of trip risk integrating minimum foot clearance and dynamic stability across the swing phase of gait," *J Biomech*, vol. 55, pp. 107–112, Apr. 2017, doi: 10.1016/j.jbiomech.2017.02.024.

[23] M. Pijnappels, I. Kingma, D. Wezenberg, G. Reurink, and J. H. Van Dieën, "Armed against falls: The contribution of arm movements to balance recovery after tripping," *Exp Brain Res*, vol. 201, no. 4, pp. 689–699, Apr. 2010, doi: 10.1007/s00221-009-2088-7.

[24] J. R. Crenshaw, K. R. Kaufman, and M. D. Grabiner, "Trip recoveries of people with unilateral, transfemoral or knee disarticulation amputations: Initial findings," *Gait Posture*, vol. 38, no. 3, pp. 534–536, Jul. 2013, doi: 10.1016/j.gaitpost.2012.12.013.

[25] J. J. Eng -David, A. Winter, A. E. Patla, J. J. Eng, D. A. Winter, and A. E. Patla, "Strategies for recovery from a trip in early and late swing during human walking," Springer-Verlag, 1994.

[26] J. C. E. van der Burg, M. Pijnappels, and J. H. van Dieën, "The influence of artificially increased trunk stiffness on the balance recovery after a trip," *Gait Posture*, vol. 26, no. 2, pp. 272–278, Jul. 2007, doi: 10.1016/j.gaitpost.2006.09.080.

[27] P. E. Roos, M. P. McGuigan, D. G. Kerwin, and G. Trewartha, "The role of arm movement in early trip recovery in younger and older adults," *Gait Posture*, vol. 27, no. 2, pp. 352–356, Feb. 2008, doi: 10.1016/j.gaitpost.2007.05.001.

[28] D. Den Hartog, J. Harlaar, and G. Smit, "The stumblemeter: Design and validation of a system that detects and classifies stumbles during gait," *Sensors*, vol. 21, no. 19, Oct. 2021, doi: 10.3390/s21196636.

[29] M. Kim and D. Lee, "Development of an IMU-based foot-ground contact detection (FGCD) algorithm," *Ergonomics*, vol. 60, no. 3, pp. 384–403, Mar. 2017, doi: 10.1080/00140139.2016.1174314.