ISR-AIWALKER: Robotic Walker for Intuitive and Safe Mobility Assistance and Gait Analysis

João Paulo, *Student Member, IEEE*, Paulo Peixoto, *Member, IEEE*, and Urbano J. Nunes, *Senior Member, IEEE*

***Abstract*—Robotic walkers are assistive robotic devices that provide mobility assistance, in a domestic or clinical scenario, to individuals suffering from a gait disorder, being age related or due to injuries, surgery, or diseases. Walkers also provide a significant potential for lower limb rehabilitation. In this paper, we present a novel multimodal robotic walker platform, the ISR-AIWALKER, where innovative contributions were made both in the human–machine interface (HMI) and in a gait analysis system placed on board the platform. Taking into account the application potential of these devices, an effort was made to use low-cost sensors without sacrificing the overall performance of the system. A change was made in the HMI paradigm, moving from a force-sensing to a vision-based approach, while maintaining a natural user interaction and adding complementary safety features like correct gripping enforcement. To cope with the close proximity of the user's body, a multimodal sensor setup was considered. Using both RGB and depth map data, a kinematic model of the user's lower limbs is obtained, allowing the identification of a set of features that are used in a machine learning approach to discriminate gait asymmetries. Experiments made with several subjects revealed that the proposed HMI is able to correctly estimate the user intention in a natural and intuitive way. The gait analysis system was also evaluated and evidenced a good discrimination capability to distinguish between different gait patterns.**

***Index Terms*—Assistive robots, gait analysis, human–machine interface (HMI), robotic walker, user intention.**

I. Introduction

MOBILITY is a fundamental characteristic of the human being. It is a natural form of exercise and locomotion. With age, mobility capabilities naturally decrease, mainly due to neurological, muscular, and osteoarticular decay. But age is not the only factor to contribute to reduced mobility. Cardiovascular complications (strokes), spinal cord injuries, diseases like multiple sclerosis, Parkinson's and others, which are not necessarily age related, also affect mobility considerably.

To help improve the lives of people afflicted by reduced mobility, the walker offers the best weight support while simultaneously promoting rehabilitation of the lower limbs. The design of such mobility aids is centered in intuitive interaction and safety. Safety is fundamental since these devices provide mobility for frail people, who frequently exhibit a lack of confidence in such assistive devices, due to a psychological condition known as fear of falling [1], [2].

This paper presents a multimodal robotic walker built resorting to low-cost technologies, where contributions are made both in the human–machine interface (HMI) paradigm, and the introduction of an innovative gait analysis system, which we developed and implemented. The motivation behind the HMI's development arose from the fact that looking into the state of the art, most of the HMIs available in this context are based on expensive technologies and partly guarantee the safety of the user, in most cases due to a lack of an approach to monitor the user's state. Safety has been typically addressed as a mere navigational problem. Most recently the user state has started to be included in the loop by analyzing the user's legs, but just from a single two-dimensional (2-D) plane perspective. Considering the rehabilitation potential of the walker, there are no works that implement onboard analysis of gait and user monitoring with the goal of providing a quantitative and qualitative gait rehabilitation system.

The proposed HMI replaces traditional force sensing technologies by a vision-based approach that allows implementing not just interaction, but also complementary safety measures, by tracking the user's hands and classifying the gripping quality to ensure a safe operation of the walker.

Considering the gait analysis system, a major contribution of this paper is the proposal of a solution able to deal with close range interaction, which is an important feature for the robotic walker's research field, but that can be extended to other research domains. We propose a multimodal vision-based gait analysis system placed on board the walker device. Resorting to several computer vision techniques, we build a kinematic model of the lower limbs. This model enables the extraction of features related to gait, which fed to a machine learning method, it is possible to classify the gait pattern, providing a way of detecting behaviors as gait asymmetries.

II. Related Work

A. HMIs on Robotic Walkers

The literature on robotic walkers presents two main approaches to HMIs. The direct HMI has the user actively exerting

Manuscript received February 9, 2016; revised September 8, 2016, November 26, 2016, February 25, 2017, and July 7, 2017; accepted September 9, 2017. Date of publication October 18, 2017; date of current version November 13, 2017. This work was supported by the project funded by FCT with reference RECI/EEI-AUT/0181/2012 and under the Ph.D. grant with reference SFRH/BD/88672/2012 with funds from QREN–POPH and the European Social Fund from the European Union. This paper was recommended by Associate Editor Manida Swangnetr. *(Corresponding author: João Paulo.)*

The authors are with the Department of Electrical and Computer Engineering, Institute of Systems and Robotics, University of Coimbra, Coimbra 3030-290, Portugal (e-mail: jpaulo@isr.uc.pt; peixoto@isr.uc.pt; urbano@isr.uc.pt).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/THMS.2017.2759807

TABLE I
SUMMARY OF THE RELATED WORK ON WALKER'S HMIS

Reference	Walker	HMI	Technology
[3], [4]	PAMM and VA-PAMAID	Direct HMI with parallel handles	3-D force/torque sensor on the center of the HMI
[5]	MARC	Direct HMI with parallel handles	2 × 3-D force/torque sensors being one for each handle
[6]	Walking Helper II	Direct HMI with forearm support	3-D force/torque sensor on the center of mass
[7] [8]	Simbiosis	Direct HMI with forearm support	3-D force sensors and ultrasounds
[9]	XR4000 platform based walker	Direct HMI with parallel handles	Force resistors on each handle
[10]	Unnamed walker	Direct HMI with parallel handles	Strain gauges on each handle
[11]	CAIROW	Direct HMI with support bar	Force resistors along the bar
[12]	ASBGo	Direct HMI with forearm support	Springs and joystick
[13]	Guido	Direct HMI with collinear handles	Force sensor and switches
[14]	JARoW	Indirect HMI	LRF extracting 2-D plane of the shins
[15]	UFES Smart Walker	Indirect HMI	LRF extracting 2-D plane of the shins and IMU placed on the user's waist

force on the walker, whereas the indirect HMI interprets user's motion to infer the desired walker's movement.

Direct HMIs typically depend on force-sensing technologies, either underused six degree-of-freedom expensive force/torque sensors, or unprecise low-cost options with low repeatability. Besides this, force-sensing technologies degrade rapidly with time. The walkers PAMM and VA-PAMAID [3], [4], MARC walker [5], the platform in [6], and the forearm support-based Simbiosis walker, which also efficiently combined feet mounted ultrasonic sensors [7], [8], implemented force/torque sensors. On the other side, we have walkers like the ones presented in [9] and [10], which used force-sensing resistors and strain gauges, respectively. Another approach, the CAIROW walker [11], equipped the support bar with several force resistors. Compared to these previously mentioned works, our approach replaces force sensing, entirely, by a vision-based solution. This allows the introduction of integrated safety measures, whereas the previous works had to implement a complementary system to address safety. Another interesting work is the ASBGo walker [12], where forearm supports are coupled to a low-cost joystick and springs, providing a safe HMI due to its body weight support topology, complemented with a body distance safety system. Comparing this approach to this paper, the use of springs is an interesting solution, serving the same purpose of avoiding force sensing and defining a neutral position. Since the user is supported by the forearms, the need for a safety measure to ensure safe gripping becomes secondary. Another well-known work presents a combination of force sensors and switches, working in different modes, in the walker Guido [13].

All of the previous HMIs require some form of gripping from the user. We consider that a mechanism to analyze the way the user grips the handles is fundamental for a correct operation of the device, namely to prevent accidents. It is noted, however, that none of the works mentioned employ any kind of safety measure to actually perceive if the user is correctly gripping the handles.

With regard to indirect HMIs, the works found in the literature are based on the use of laser range finders (LRF). Some examples of this paradigm are the JAIST Active Robotic Walker (JARoW) in [14] and the work in [15], which combines one LRF with inertial sensors placed on the user's waist. These indirect interfaces provide an interesting alternative; they employ LRFs to detect the legs' positions in the provided 2-D plane, operating consistently no matter the gait pattern. However, an important aspect to consider is the fact that in certain scenarios, like for instance Parkinson's disease, the users require an active propulsion to start walking, a phenomenon known as freezing of gait [16]. With these kinds of indirect interfaces alone, it is impossible to provide such assistance due to the fact that an initial leg motion is required. However, the use of complementary technologies, like voice commands to order a motion from the walker, has been used to overcome this limitation. Table I summarizes the works mentioned and provides an overview of the current HMIs on walkers. A more extensive overview on robotic walkers is given in [17] and [18].

B. Gait Analysis on Robotic Walkers

The robotic walker is primarily a locomotion device, but it has also a high rehabilitation potential. As such, it is important to understand how this topic of gait analysis was addressed in the robotic walker context. We are particularly interested in approaches that can be deployed on board the robotic device. Approaches like the works in [19]–[21], based on multiple fixed cameras, or the use of body inertial sensors as in [22], are excluded for being unpractical in this scenario.

Looking to the state of the art, we conclude that there is the lack of a solution for gait analysis on board the walker. Some recent works, still preliminary, try to address this using RGB-D cameras on board, but there is still the problem of the minimum effective range of the camera since the sensors used so far are either Microsoft's Kinect or Asus Xtion, sensors designed for a room. Examples are the works that take into consideration only feet and shins, modeling them for a future disorders diagnosis application [23], or pose estimation [24]. In [25], the approach focuses just on the feet pose estimation.

These new approaches are an interesting step forward. The use of low-cost sensors is a benefit and the change of paradigm to an RGB-D context allows a complete perception of the user's body parts. However, we consider that these approaches are still in a preliminary state since only small parts of the lower limbs are perceived. Although these previous works focus on the perception and estimation of lower limb poses, they do

TABLE II
SUMMARY OF THE RELATED WORK ON GAIT ANALYSIS SYSTEMS

Reference	System	Technology
[19], [20]	Multiple cameras facing same area from different perspectives	Vision based with markers
[21]	Multiple cameras facing same area from different perspectives	Vision based and markerless
[22]	Multiple inertial sensors placed on the user	IMUs
[23], [24]	Single RGB-D camera on board the walker	Microsoft Kinect
[25]	Single RGB-D camera on board the walker	Asus Xtion

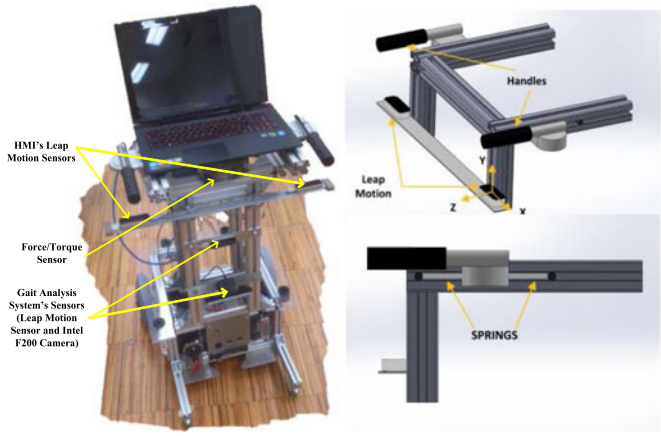

Fig. 1. Robotic walker platform and HMI's schematic.

not provide mechanisms to classify gait patterns or diagnose disorders, nor do they provide a tool to assess rehabilitation progress. The previously mentioned works are also grouped in Table II.

III. SYSTEM OVERVIEW

A. Robotic Walker

The robotic walker used in this paper was built entirely in our lab (see Fig. 1). It is composed by a differential motorized platform with two rear castor wheels. The platform was designed to drive a nominal load at 80 kg, while structurally it can support higher loads. It is equipped with two motors from Maxon, integrating a gearhead and encoder on each. The motors are driven by a power controller from Roboteq, on which we designed and implemented a PID controller for a fast response from the motors. The platform is powered by lithium-ion batteries, which are managed by a Battery Management System also developed in our lab.

The upper structure of the robotic walker, dedicated to the interaction with the user, integrates the HMI and gait analysis systems along with a force/torque sensor of six degrees-of-freedom from ATI. The sensor is used for the sole purpose of aiding in the HMI's development.

B. HMI's Architecture

Our HMI follows an architecture similar to the conventional walker, with a parallel handle configuration. The sensor technology employed replaces the use of the traditional approach with force sensing technologies. This solution is inexpensive and does not degrade with time. The sensors used are the vision-based hand tracker device from Leap Motion. The Leap Motion sensor is equipped with two infrared (IR) cameras and three IR light-emitting diodes to illuminate the scene. The side-by-side cameras capture the scene at 100 Hz. The sensor outputs the position of each hand's points relative to its reference. In the proposed setup, two sensors were used one for each handle, positioned under them at a distance of 15 cm and perpendicular to the handle's longitudinal axis (see Fig. 1).

Since the Leap Motion sensor is vision based, the interaction of the user with the interface requires a displacement of the handles to infer a command. Each handle will slide forward and backward, returning to the center position, due to the use of springs. The sensors capture this sliding displacement when the user grasps the handles. The stiffness of the springs is dimensioned to offer a small resistance, giving the impression of pushing a light device.

The displacement of the hands when gripping the handles is directly related to the compression distance of the springs, making it possible to determine the user's push or pull force and infer a continuous range of commands instead of simple binary ones (forward, stop, turn right, etc.). The schematic representation of the proposed interface is shown in Fig. 1.

The command signal inputted to the system is the displacement of each hand given by the respective sensor, specifically the Z-axis variation of each hand's palm position, when the user grasps the handles, which is the parallel axis to the handle's displacement, as shown in Fig. 1. The development of the HMI is detailed in Section IV.

C. Gait Analysis System Setup

The gait analysis system takes advantage of recently commercialized sensors, which are employed for different purposes, but here they were resourcefully adapted for gait analysis. The use of sensors like the Kinect or similar (Asus Xtion) is impractical in this scenario since the user's body is at a very close proximity to the walker's frame (<300 mm). The above-mentioned sensors have a minimum effective range of at least 500 mm.

Considering the options available in the market and taking just into consideration vision-based approaches with short minimum ranges, the first sensor chosen was the Leap Motion controller. As previously described in Section III-B, this sensor was developed specifically for hand tracking, but since it can provide the raw data from both of the IR cameras, we can use it like a normal stereo system. With a minimum effective distance of 25 mm, it is suitable for short-range object detection and analysis. The other sensor integrated into the setup is an RGB-D camera from Intel, the F200 front facing camera, commercialized for face and gesture recognition. It is a structured light-based camera, like Microsoft's Kinect, but smaller and with minimum effective distance of 150 mm.

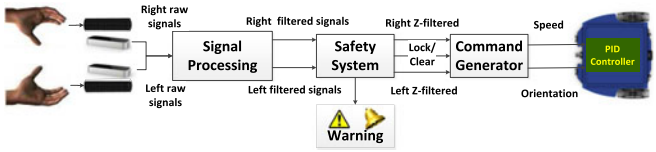

Fig. 2. HMI's flowchart.

The purpose of each sensor in this setup is better explained with the help of Fig. 1, which shows the system's setup on board the robotic walker. The Leap Motion controller is pointing toward the middle of the user's legs. Its purpose is to capture waist and legs, allowing the kinematic modeling of the lower limbs. The data from feet and ankle are captured by the F200 camera. It allows heel strike detection and the extraction of other features for gait analysis. The use of several sensors is required due to the close proximity of the user's body. Using just one sensor is not enough to capture waist, legs, and feet. How the data from the sensors are combined for gait analysis is explained in Section V.

IV. Human–Machine Interface

This section describes the structure of the HMI implemented on the walker, outlined in Fig. 2. As mentioned before, the input signal is the position of the hands of the user given by the Leap Motion sensors. Looking at Fig. 2, three modules are responsible for the HMI's operation. The first module filters the Leap Motion sensors' signals to remove noise and outliers, as soon as the user starts to perform a reach-to-grasp motion. The second module is responsible for guaranteeing that the user is correctly gripping the walker's handles. This module determines if the performed reach-to-grasp pattern is safe for the operation of the walker, providing the "clear for operation" signal. If safety conditions are not met, the motors are locked and an auditory cue is triggered. The last module, the interface's command generator, is responsible for the translation of the hands' displacements into a range of navigational commands. These modules work continuously in a loop. The HMI's conceptual design was previously presented in the work [26].

A. Inferring User Intention

The purpose of any HMI is to translate an intention, which is expressed by an interaction behavior or action, into a command. This interaction should be as intuitive as possible for a good acceptance from the user. We performed a study during the interface's design to determine the most suitable combination of handles' displacements that immediately relate to specific maneuvers (navigational commands). This allowed us to focus the design of the HMI toward an intuitive and accurate user intention determination.

First, we aimed to understand the correlation between the forces/moments exerted by the user on the walker with the corresponding linear velocities and curvature angles involved in each walker maneuver (turn left, walk forward, etc.). The next step was the analysis of the correlation between the relevant forces on each maneuver and the handles' displacements. The linear velocity and curvature angle were calculated from the motor encoders at every sample, considering that the platform is actually a differential robot.

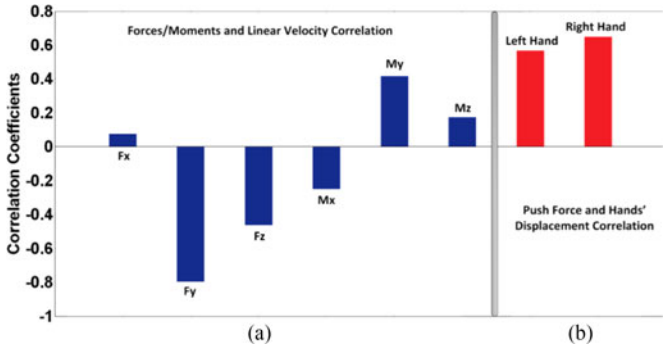

Fig. 3. Correlations for a forward maneuver. (a) Between forces/moments and linear velocity. (b) Between push force (Fy) and hands displacements.

Five healthy volunteers between 25 and 35 years old were invited to collaborate in this experiment. They were asked to perform specific maneuvers with the walker, such as walk forward, turn left/right 90°, and walk backward. The device was passive at this point, which required the user to maneuver the full weight of the device. A dataset was collected and all the postprocessing was done offline.

Considering the force/torque sensor shown in Fig. 1, its coordinates system is composed by the up vertical axis, which is the positive Z, followed by the forward positive Y (pointing toward the front of the walker), and right positive X. The results from this experiment revealed that for a linear motion the force in Y is highly correlated with the linear velocity and poorly correlated with the curvature angle. For forward and backward maneuvers, the set of handles displacements is characterized by both handles being either pulled or pushed toward or from the user, respectively. In the case of sharp turns, the relevant variable is the moment around the Z-axis, which is highly correlated with the curvature angle and poorly correlated with the linear velocity. Both handles slide in opposite directions, depending on the side of the turn.

As an example, Fig. 3 represents a forward maneuver from a standing still point to a total halt, showing the correlations between forces/moments and the linear velocity, as well as, the observed relevant force for this specific maneuver, the push force (Y-axis), and the handles' displacements. As a result of this experiment, we obtained a set of decision-making rules that are integrated into the fuzzy-logic command generator.

B. Command Generator

The command generator module is responsible for converting the handles' displacements into motor commands. The control is based on fuzzy logic, which follows the premise that precise outputs can be obtained from imprecise or vague inputs [27].

The proposed controller is based on the following operators: Takagi–Sugeno fuzzy inference, Gaussian fuzzifier, minimum t-norm, Mamdani minimum implication, maximum aggregation, and centroid defuzzifier.

TABLE III
DECISION-MAKING RULES FOR THE FUZZY LOGIC COMMAND GENERATOR

		Right Hand					
		F		N		B	
Left Hand	F	Front	Straight	None	Straight	None	Right
	N	None	Straight	None	Straight	None	Straight
	B	None	Left	None	Straight	Back	Straight

The inputs of the controller are the left and right Z-axis hands' displacements, described by the linguistic terms $T_{\text{left}} = T_{\text{right}} = \{A_1,\ A_2,\ A_3\}$ (Backward, Neutral, and Forward) with universe of discourse of [–3, 3][cm]. The fuzzy membership functions associated to $A_1,\ A_2,\ A_3$ are defined by Gaussian shapes.

The outputs of the controller are the linear and angular speed commands. The linear speed command is described by the linguistic terms $T_{\text{linear}} = \{A_1,\ A_2,\ A_3\}$ (Back, None, and Front) with universe of discourse of [–100, 100][%]. The fuzzy membership functions associated with $A_1,\ A_2,\ A_3$ are also defined by Gaussian shapes. On the other hand, the angular speed command is described by the linguistic terms $T_{\text{angular}} = \{A_1,\ A_2,\ A_3\}$ (Left, Straight, and Right) with a universe of discourse of [–100, 100][%]. The fuzzy membership functions associated with $A_1,\ A_2,\ A_3$ are defined by Gaussian shapes.

The decision-making rules for the system are summarized in Table III. The outputs of the controller are a range of commands, which means that the higher the force exerted by the user on the handles the higher the compression of the springs and as such the longer the displacement of the handles will be. This progressive displacement will result in continuous speed command curves rather than discrete speed step commands. We also define a dead zone, where the user's motions do not cause involuntary motor commands to prevent accidents when the user stops to rest or when the user performs the initial hands' approach and gripping adjustments.

The outputs of the fuzzy command generator serve as the inputs for the PID controller that directly drives the walker's motors. The controller's smooth acceleration and deceleration ramp serves as the final signal processing stage.

C. Gripping Safety System

Safety is a critical factor when it comes to mobility aids, especially in a walker platform, in the sense that it requires body-weight support through hand gripping. To ensure a safe operation of the robotic walker, a system to detect if the handles are being correctly gripped (both hands' palms facing each other) was developed. The correctness of the preceding reach-to-grasp gesture indirectly ensures the correct gripping pattern. Each gripping is classified as suitable for operation or as unsafe with a probable risk of fall.

The safety system follows the functional diagram of Fig. 4. The system has a higher priority than the command generator, only allowing the robotic walker to move if safety conditions are met. The following paragraphs describe the development of

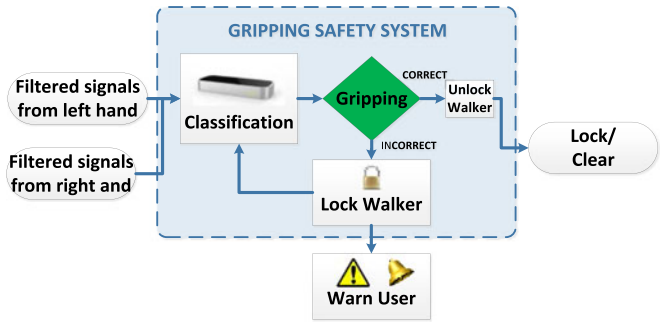

Fig. 4. HMI's safety system's functional diagram.

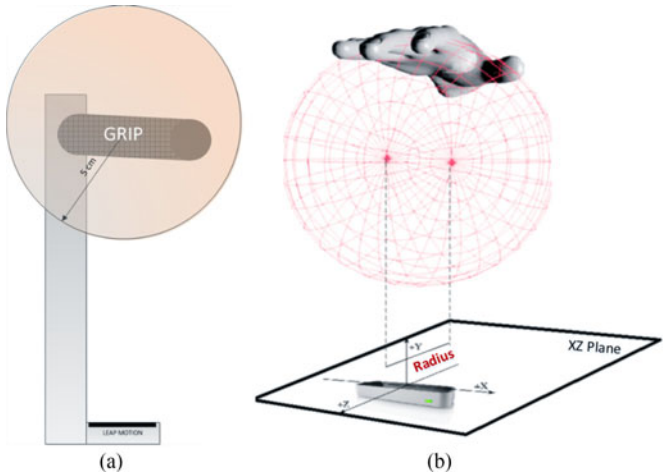

Fig. 5. Safety system's (a) gripping region of interest and (b) hand's sphere radius feature.

the safety system previously proposed in [28], and now implemented in this new robotic walker. We did not opt for a contact solution like a resistive or capacitive sensor, due to the fact that these sensors do not provide enough data to discriminate between gripping patterns neither do they have the capability to capture the reach-to-grasp gesture in all its trajectory. In this sense, the Leap Motion controller provides an all-in-one solution.

When the hands of the user are detected entering a defined region of interest, some discriminative features are extracted. These features are fed into a previously trained discriminative classifier that handles the classification of the reach-to-grasp gesture. We considered that the region of interest for each grip, where the reach-to-grasp gesture is going to be detected and classified, is the spherical space with 5 cm of radius around the center point of the grip, as shown in Fig. 5(a). The hand is assumed as being inside the region of interest when its center point is inside that spherical space. All points refer to the sensor's coordinate reference system.

For the classification, a set of features is extracted at every frame. Those features are the following: the radius of the sphere that fits the curvature of the hand, the palm position, and the palm's unitary direction vector. The selection of these features was considered to completely define the shape and movement of the hand at each collected sample.

Fig. 6. Gait analysis system's functional diagram.

For the calculation of the sphere's radius, a technique of point projection was considered. The palm's center point and the middle finger's tip are projected into the sensor's *XZ* plane, as shown in Fig. 5(b). The distance between these two points in this plane is the radius of the sphere that fits the hand's curvature.

The second feature that was considered was the center point of the palm. The palm's position is automatically provided by the sensor. This point was also used for the calculation of the last feature, the palm unitary direction vector. The vector is defined by a middle finger's base point, which separates palm from finger, and the palm's center point, as in

$$\vec{d} = \frac{< F_x - P_x, F_y - P_y, F_z - P_z >}{\sqrt{(F_x - P_x)^2 + (F_y - P_y)^2 + (F_z - P_z)^2}} \quad (1)$$

where $F_{(x,y,z)}$ is a middle finger's base point and $P_{(x,y,z)}$ is the palm's center point. The resultant vector is the palm's unitary direction vector.

For classifying the gripping quality, we adopted a supervised learning model. The goal was to recognize the different gestures and get a binary output. The binary output was divided into adequate or inadequate gripping. The classifier used was a support vector machine (SVM) [29].

V. Gait Analysis System

The gait analysis system is composed by several modules combined to achieve the goal of lower limb detection and gait pattern classification. We defined the gait analysis functionality only to operate when linear motion is occurring, meaning rotational motion is not used for gait analysis purposes. The functional diagram is schematically represented in Fig. 6. This section presents a detailed description of the techniques used to develop the gait analysis system, taking advantage of the proposed sensor setup.

A. Understanding Human Gait

To develop an efficient and usable gait analysis system, first we need to understand human gait, specifically its phases and the relevant body part components involved in the process.

In the medical and rehabilitation fields, the heel strike is used to divide the gait cycles [30]. Each heel strike on the ground separates the left and right strides. In [30], a study was performed to assess which were the relevant parameters needed to be monitored in order to fully characterize a gait cycle. Both a spatiotemporal and a joint excursion analysis of the patient's legs were performed. The conclusion was that the best parameters that should be considered for the analysis of the gait cycle (from heel strike to heel strike) are gait speed, step length and frequency (spatiotemporal parameters), and parameters related to the hip, knee, and ankle joints (kinematic parameters). These considerations guided the development of our gait analysis system.

Since the target population of walkers is individuals suffering from gait impairments, gait analysis for evaluation of such impaired gait persons needs to be compared with data from healthy people. Walkers are intended to operate either in a domestic scenario, as in the case of elderly people, or in a clinical scenario, as in the cases of surgery recovery or rehabilitation therapies, due to injuries and degenerative diseases. The studies in [31] and [32] provide reference data for spatiotemporal and joint angle parameters. These studies reference data from 233 healthy subjects at different gait speeds. Another study [33] provides gait parameters data obtained in a walker-assisted scenario with disabled subjects. This paper is very helpful to provide an insight on how gait parameters are influenced in such cases. From these studies, conclusions are that between normal and assisted gait the most relevant difference is a shorter step length in the latter case and that the overall spatiotemporal parameters all suffer a reduction. Furthermore, in healthy individuals no differences were observed between left and right stride; this means that a healthy gait does not present an asymmetrical pattern.

Gait analysis, however, does not just relate to lower limbs tracking. In fact, the upper body is also relevant. The analysis of the trunk's movements and posture as well as the shoulders alignment provides relevant information when facing free ambulation. In this paper, we did not tackle upper body monitoring, due to the fact that the user is supporting his body weight on the walker's handles. However, we still consider it to be of some relevance even in our case, and we will address it in the near future work.

B. Feet and Heel Strike Detection

From the previous section, it is clear that the heel strike detection should be the first step for gait analysis. It provides the separation point between each stride and defines the window where feature extraction to classify the gait pattern should occur. The sensor that monitors the feet is an Intel F200 camera. The data that will be considered are both the RGB image and the provided depth map (DM). The method proposed to achieve heel strike detection is described next.

Fig. 7. Feet and heel strike detection methodology's techniques. (a) Represents the raw data from the Intel F200 camera, (b) is the result from the floor subtraction, (c) demonstrates the result of the clustering technique to extract the feet region, and (d) shows the accumulation vector resultant from the circular Hough transform with a temperature gradient.

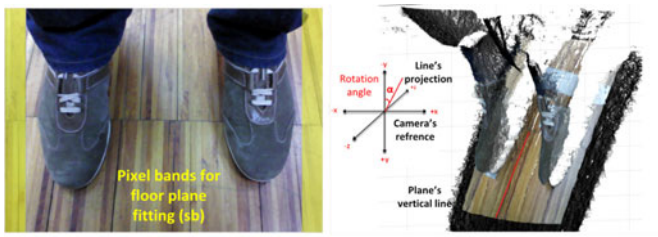

Fig. 8. Visualization of the parameters for the floor subtraction method.

Algorithm 1: Floor subtraction.

Input: $RGB = (u, v)\,, DM = (x, y, z)$

$sb : sb \leftarrow$ extracted side bands from DM, Fig 8 a)
$floorplane := fitplane\,(sb)\,;$
$\mathrm{l}_y : \mathrm{l}_y \leftarrow$ projected plane's line on the Y-axis, Fig 8 b)
$\alpha : \alpha = tan^{-1}\,(slope(l_y))\,;$
$DM_{rot} := rotate\,(DM, \alpha, camerareference)\,;$
foreach DM_{rot} *pixel* **do**
  if $(DM_{rot,z} > dist(floorplane, imageplane)$ **then**
    $DM_{rot,(x,y,z)} = \mathrm{Null}$
$DM_{clean} := rotate\,(DM_{rot}, -\alpha)\,;$
foreach RGB *pixel* **do**
  if $(DM_{clean,z} == \mathrm{Null}$ **then**
    $RGB_{(u,v)} = 0$

Output: $RGB; DM_{clean}$

1) Raw Data: Fig. 7(a) shows the RGB image and DM obtained from the Intel F200 camera. It gives a perspective of the raw data collected from this sensor. Each pixel of the RGB image corresponds to the same pixel in the DM.

2) Floor Subtraction: As a normal technique in computer vision, the background subtraction allows separating the relevant data from the rest of the environment. In our context, the relevant data are the feet and the background is the floor plane, which is at a constant distance from the camera and at a certain inclination angle. The expected result is to have an image with only the feet and part of the leg, as shown in Fig. 7(b). To reduce the computational cost, this step is done in the first frame and the parameters are used in the subsequent ones.

The process is divided into several consecutive steps enumerated by Algorithm 1, which uses the RGB image and DM, as shown in Fig. 8.

3) K-Means Clustering: This clustering is performed on the RGB image, with the intention of getting a separation between the feet and the legs by color clustering. The final RGB image will only contain pixels from the region of the feet, the remaining pixels being cleared. This technique is used on the assumption that the footwear color distribution is most probably distinguishable from pants, socks, or even skin.

The *K*-means clustering is an unsupervised clustering method. It classifies data points into classes based on the distance from each other. This distance is taken depending on the chosen metric. Points are clustered around centroids $\mu_i, \forall i = 1, \ldots, k$, which are found by minimizing the objective

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2 \tag{2}$$

where k is the number of clusters and μ_i is the ith centroid. In our case, we considered two clusters and a metric based on the Euclidian distance. The result from this processing step is visible in Fig. 7(c).

4) Circular Hough Transform: The circular Hough transform is an extension of the Hough transform specialized in finding circles in input images. The circle candidates are obtained from a voting process in the Hough parameter space, followed by finding the local maxima from an accumulator matrix. We propose this technique due to the fact that the footwear's tip shape typically resembles a semicircle.

This technique is used in the RGB image resultant from the previous step. By applying the circular Hough transform, we obtain the accumulator matrix, which is a matrix with the dimensions of the RGB image, where each pixel has a value corresponding to the number of circles that pass through it. Finally, we consider the two global maximum points of the accumulator, which will typically correspond to the feet's tip's center point. Fig. 7(d) shows the accumulator matrix where the brightest points in the image are the maximums, allowing the visual perception of the feet detection algorithm's result.

5) Heel Strike Detection: In this step, the global maximum points from the accumulator matrix obtained in the previous step are mapped into the DM. The detection of the heel strike is a time-dependent process that will basically monitor the depth variation of the closest foot to the robotic walker (see Fig. 9). A heel strike event occurs when the considered foot depth transitions from a rising state to a descent state. Algorithm 2 describes the process followed every frame for detecting a heel strike. To prevent unreliable data from the furthest foot disappearing from

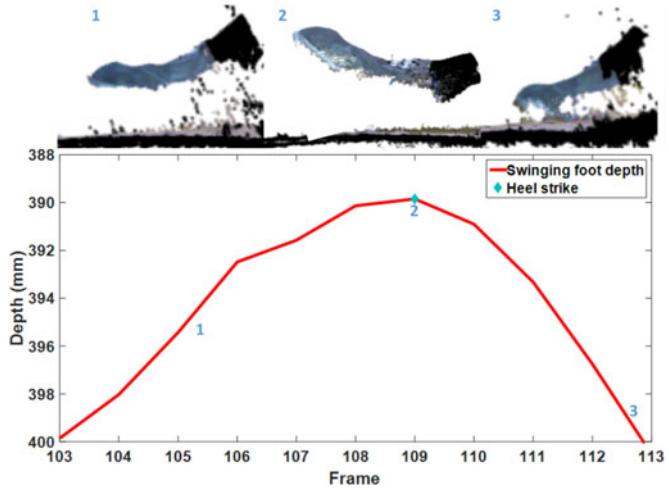

Fig. 9. Heel strike detection during the swinging's foot depth variation (sagittal plane view).

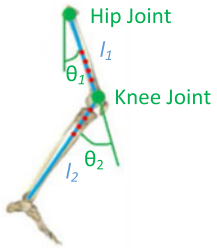

Fig. 10. Kinematic model for each lower limb where red dots represent fixed distance points to the knee joint.

Algorithm 2: Heel Strike Detection.

Input: $Foot_{left} = (x, y, z)$, $Foot_{right} = (x, y, z)$

if $(dist(Foot_{left}, Foot_{right}) < 0))$ **then**
  if $(findpeaks(Foot_{left}, [T-1:T])$ **then**
    $\text{Heel}_{left}\_T := T$; ← left heel strike event
else if $(findpeaks(Foot_{right}, [T-1:T])$ **then**
  $\text{Heel}_{right}\_T := T$; ← right heel strike event

Output: $\text{Heel}_{left}\_T, \text{Heel}_{right}\_T$

the camera's field of view (FOV), we define a boundary in the *Y*-axis of the DM. If a foot's tip position crosses that boundary, we keep its last boundary position (*x,y,z*) until it re-enters the region of interest.

C. Lower Limbs' Kinematic Model

The sensor responsible for the perception of the lower limbs is the Leap Motion sensor. It is placed facing the coronal plane of the user (frontal view), but the data are handled so that it is possible to have a projection in the sagittal plane (side view). The lower limbs or legs are modeled as a two-link revolute robot, seen from the sagittal plane perspective. The model has two links, with the hip and knee as two joints, as shown in Fig. 10. To build this kinematic model, the following sequence of steps is required.

Algorithm 3: Legs segmentation.

Input: $I_c = (u, v)$

$L_{cp} : L_{cp} \leftarrow \{\}$set of v-coordiantes of left leg's between contour points
$R_{cp} : R_{cp} \leftarrow \{\}$set of v-coordiantes of right leg's between contour points
$N : N = 0$;
foreach row of I_c **do**
  if $(numberofcountours == 4)$ **then**
    $L_{cp} := L_{cp} \cup \{middle(I_c(i, c(1)), I_c(i, c(2)))\}$;
    $R_{cp} := R_{cp} \cup \{middle(I_c(i, c(3)), I_c(i, c(4)))\}$;
    $N++$;
$L_{cp\_final} := \sum L_{cp}/N$
$R_{cp\_final} := \sum R_{cp}/N$

Output: $L_{cp\_final}, R_{cp\_final}$

1) Raw Data and 3-D Point Cloud Reconstruction: The objective of this first processing step is to obtain a 3-D point cloud of the legs. To achieve this, we use the Leap Motion's stereo images, as shown in Fig. 11(a). We handle these images as a normal stereo system. We calibrate both IR cameras [34], rectify the stereo pair, compute the disparity map through a semiglobal matching technique [35], and finally reconstruct the 3-D scene [see Fig. 11(b)].

2) Leg Segmentation: In this step, both legs are segmented and modeled as two vertical lines that are equidistant to each leg's contours in the frontal plane (left and right contour). Using the base image described in the previous step, a Canny edge detector is used to extract the contours of the waist and legs. Fig. 11(c) shows an example of the resulting waist and legs contour. The process of extracting each leg's vertical line consists in computing the set of center points between the left and right contour for every horizontal line in the leg area [points c_i in Fig. 11(c)].

This set is then used to compute the average *v*-axis coordinate, which will define the vertical line that passes through the leg. Algorithm 3 describes this process which is computed at every frame.

The points on those two lines are mapped into their corresponding 3-D points using the stereo calibration information. Their projection in the *ZY* plane (sagittal plane) corresponds to the legs' surface profile [red dots in Fig. 11(d)]. A limitation of this approach is that for individuals with joint deformations, which cause each leg's frontal view not to be modeled resorting to a vertical line, the result of the legs' segmentation can become unreliable.

3) Knee Detection: To build the kinematic model of the lower limbs, it is imperative to find the position of the knee (one of the model's joints). From the previous step, we determined a vertical line that passes through the center of each leg. Assuming that this line passes through the knee, we can concentrate our attention on the 3-D point cloud points that belong to the sagittal plane that contains this line. These points will provide us a sagittal perspective of each leg [see Fig. 11(d)].

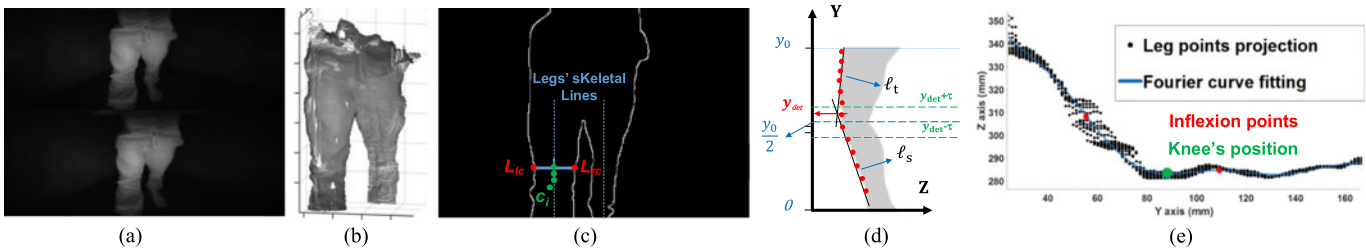

Fig. 11. Lower limbs' modeling method. (a) Presents the stereo raw data, (b) 3-D point cloud reconstruction, (c) waist and legs contour where the skeletal line of each leg is determined, (d) knee's y-interval estimation, and (e) shows the fitting to the leg's point cloud interval projection to determine the vertical knee point.

Considering the projection of each leg in the ZY plane, we initially estimate an interval of values in the Y-axis as the probable interval for each knee's position Y-value. As shown in Fig. 11(d), we divide each projection into two halves $y_0/2$. For each half, we fit a line using least squares. The intersection of the two lines provides the y_{det} point of each knee. The interval is defined by the points in the gate $(y < y_{\text{det}} + \tau) \wedge (y > y_{\text{det}} - \tau)$, where τ denotes the limits of the gate.

Next, we fit a fourth degree Fourier curve to each interval of the projections to smooth the result. This previously determined interval region contains the knee cap's dome-shaped points. Analyzing the resulting curve for each leg, as exemplified in Fig. 11(e), we realize that the knee's position lies between two inflection points of the fitted curve. To determine this position, the second derivative of the curve is computed. The knee's position is the global minimum between the inflection points. The output of this step is the point position of each knee in the respective projections.

4) Kinematic Model Fitting: The kinematic model employed is a two-link robotic arm model (thigh and shin) with two joints (hip and knee). This model is fitted into the 3-D points that belong to the sagittal plane described in the previous step. The basic problem that needs to be solved is the identification of the angles of each leg joint. Considering Fig. 10, the forward kinematics equations that rule this model are described by

$$x = l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)$$
$$y = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2). \quad (3)$$

To minimize the complexity of the problem and looking into Fig. 10, specific points were chosen along the two links. The angles are determined by an iterative process of error minimization of the respective model points' positions against the considered points' positions in the sagittal plane. The error function is given by (4), where N is the number of fitting points used, and the goal is to minimize it taking into account several combinations of joints' angles as in

$$E(x) = \frac{1}{N}\sum_{i=1}^{N}(x_{i\text{model}} - x_{i\text{sensor}})^2, i = 1, \ldots, N$$
$$E(y) = \frac{1}{N}\sum_{i=1}^{N}(y_{i\text{model}} - y_{i\text{sensor}})^2, i = 1, \ldots, N \quad (4)$$
$$\min_{\theta_1,\theta_2} E(x) + E(y). \quad (5)$$

TABLE IV
SELECTED FEATURES FOR GAIT CLASSIFICATION

Feature	Definition
Step length (at heel strike)	$\text{dist}(\text{foot}_{\text{left}}, \text{foot}_{\text{right}})$
Step time (hs to hs)	$T(\text{hs}^T) - T(\text{hs}^{T-1})$
Δ, μ of hip's joint's angle (hja) of swinging and support leg	$\Delta = \max(\text{hja}) - \min(\text{hja})$, $\mu = \frac{\sum_{i=\text{hs}^{T-1}}^{\text{hs}^T} \text{hja}}{N}$
Δ, μ of knee's joint's angle (kja) of swinging and support leg	$\Delta = \max(\text{kja}) - \min(\text{kja})$, $\mu = \frac{\sum_{i=\text{hs}^{T-1}}^{\text{hs}^T} \text{kja}}{N}$
Δ of knees baseline separation	$\Delta = \text{dist}(\text{knee1}_{\text{hs}}, \text{knee2}_{\text{hs}})$
Maximum swinging foot's rise (foot_z)	$\max(\text{foot}_z)$
Δ of the swinging foot's rise	$\Delta = \max([\text{foot}_z]_{\text{hs}-1}^{\text{hs}}) - \min([\text{foot}_z]_{\text{hs}-1}^{\text{hs}})$

D. Gait Classification

More than just detecting and recording specific lower limbs' features, the gait analysis system is intended to also automatically recognize some gait patterns, in order to help healthcare professionals to reach a diagnostic of certain diseases and track rehabilitation therapy progress. This system is a tool for diagnostic and not a diagnostic provider. The use of machine learning techniques allows the identification and classification of specific gait patterns. In this paper, we propose a machine learning framework with the objective of detecting asymmetric gait patterns. A set of features is identified and used to discriminate different gait patterns. The following description addresses the features and classifiers considered in our approach.

1) Feature Extraction: From the analysis of Section V-A, we selected several relevant spatiotemporal gait and joints' parameters, obtained from both legs during each heel strike (hs). These parameters and combinations between them will serve as features that will be used for gait classification. These features are obtained from the feet detection method and from the kinematic model. The considered features are summarized in Table IV.

2) Classification Strategy: In this paper, we used a classification strategy that could efficiently classify gait in terms of normal gait and asymmetric gait. In an asymmetric gait, right and left stride present different variations in the gait parameters. Thus, each stride is classified against two possible classes,

normal stride or deficient stride. An asymmetric gait pattern is identified when at every two strides, one is classified as deficient.

The classifier used is a binary-class SVM, which we implemented using the LibSVM package [36]. The feature set is normalized to guarantee that the minimum and maximum values obtained during the training stage were applied on the testing set. The SVM was trained according to the strategy "one-against-all," using a soft margin (cost) parameter set to 1.0. The chosen kernel is the cubic kernel because it was the kernel that provided the best results during the experimental evaluation that was performed.

E. User's Stability Safety System

Taking advantage of the lower limb's kinematic model, we developed a system to check if the user's issued motion commands are consistent with the user's pose. This system brings together the HMI and the gait analysis modules, adding an extra safety feature to the walker.

When a motion command is issued by the user, we leverage the extracted feature of the user's detected knees' positions to determine if the user is operating the walker within a safe region (between $\text{dist}_{\min}$ and $\text{dist}_{\max}$). The average position of the user's knees is considered to determine the user's distance to the walker, as given in (6). If this position is not within the safe region, the walker is stopped to prevent hazardous situations

$$\text{dist}_{\min} < \text{dist}(\text{average}(\text{knee}_z^{\text{left}}, \text{knee}_z^{\text{right}}), \text{walker}) < \text{dist}_{\max}. \tag{6}$$

VI. Results

A. HMI's Safety System Evaluation

The first experiment was designed to evaluate the performance of the classifier in discriminating between an adequate gripping and an incorrect one. For this experiment ten volunteers collaborated, eight males and two females with ages varying from 25 to 35 years.

Subjects were asked to repeat a sequence of reach-to-grasp gestures. Three types of reach-to-grasp gestures were considered. The first one was the correct way of reaching and safely gripping the walker, whereas the other two were labeled as inadequate and potentially dangerous. Each subject faced the walker at a reaching distance of 20 cm and performed 20 repetitions for each reach-to-grasp gesture with both hands simultaneously.

We adopted a leave-one-out cross-validation test. The learning was made with eight persons and tested with two unseen persons. The classification was made for each individual frame to account for the accuracy of the frames correctly classified. To evaluate the performance of the classifier, we present the precision–recall curve as well as the receiver operating characteristic (ROC) curve in Fig. 12. Looking at the ROC curve, we calculate the quantitative parameter area under the curve (AUC), which is 0.9869. The AUC represents the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one. Other classification results can be found in Table V, such as the overall accuracy of 94.76%.

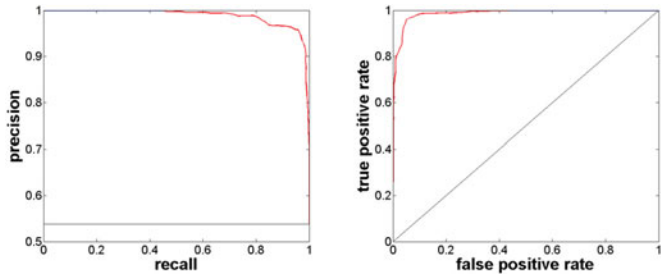

Fig. 12. Precision–recall and ROC curves for the safety system's classifier.

TABLE V
PERFORMANCE RESULTS OF THE HMI'S SAFETY SYSTEM

	Accuracy (%)	Precision (%)	Recall (%)
Classification	94.76	96.40	93.77

Regarding limitations of the system, we verified that fast and sudden erratic reach-to-grasp motions result in the inability of the system to correctly track the motion, leading to incoherent classification results. Such limitation may be an issue when dealing with persons suffering from upper limbs motor coordination problems. For fist-shaped gestures, we verified that the finger tracking is compromised, as is the response from the classifier.

In general, results show a satisfactory generalization of the classifier. The incorrect gripping trajectories are detected with high accuracy and the system prevents any walker operation. It is worth mentioning that because both handles are classified independently, the user can perform nonsimultaneous reach-to-grasp gestures with each hand. However, both gestures need to result in correct gripping patterns to be classified as being adequate for operation. A possible extension to our approach might be the use of some form of force sensing in the handle to provide information regarding how tightly the user is gripping it.

B. HMI's Operation Evaluation

To validate the proposed HMI, we conducted an experiment with five healthy volunteers. We wanted to evaluate the extent to which the proposed system would provide active assistance to users by helping them to perform the desired maneuvers, and to understand if the HMI could cope with weight-bearing conditions, which is an essential attribute of this aid for disabled individuals. We also collected the subjects' feedback about their experience with the walker through a qualitative inquiry. The subjects were asked to maneuver the walker on a predefined track, as shown in Fig. 13. The shape of the track involved maneuvers that completely characterize the HMI's span of operation. We collected data of the walked path from the motors' encoders, force/torque sensor, and handle displacements.

The results of this experiment showed that the volunteers were able to accurately navigate through the proposed path without the need to stop to adjust their current trajectory, while being able to walk at the intended speed, as observed by analyzing both the trajectories and the handles' displacements of Figs. 13 and

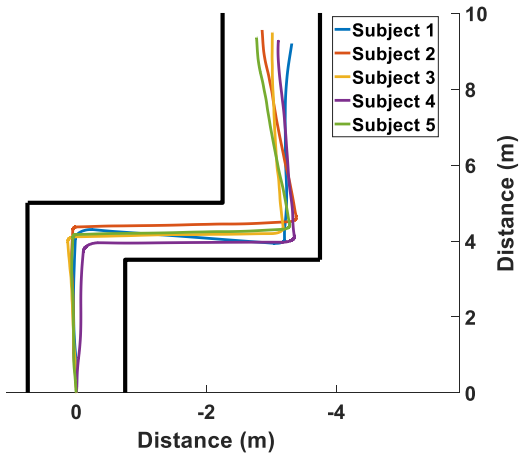

Fig. 13. Performed trajectories during the experimental tests.

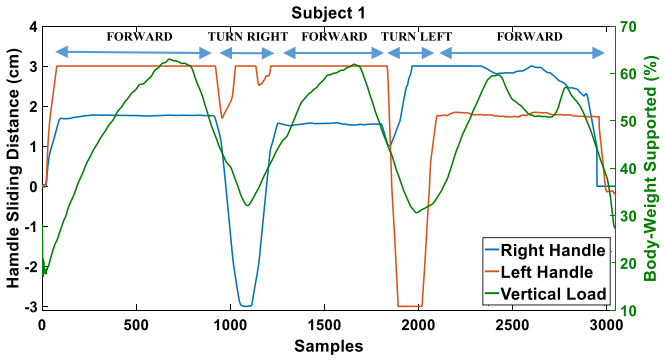

Fig. 14. Subject 1's body-weight and handles' displacements on the path.

14. The subjects were able to navigate along the proposed path, imposing on the walker a considerable body weight percentage, while being able to slide the handles smoothly and accurately. As an example, we have the curves' profile in Fig. 14; here, the subject walked down the path supporting a considerable amount of body weight on the platform and was able to maneuver it as expected.

After the experiment, we received positive feedback from the users about their maneuvering experience. All the subjects revealed that the interaction was very intuitive, allowing the platform to be easily maneuvered while supporting their body weight. Furthermore, the walker's motion was in sync with the users' intents and provided a sense of security.

We remark that the interface, although operating satisfactorily with healthy subjects, can see its performance hindered when dealing with users evidencing low dexterity, such as advanced Parkinson's patients. As such we note that the HMI operates at its best with individuals that need to support their weight, but which maintain some upper limb control capabilities, without erratic tremors. Although the test was performed with the individuals walking at their intended speed, further testing will allow to understand the sensitivity of the HMI in terms of speed control. This is particularly important when testing the system using impaired individuals.

C. Gait Analysis System Evaluation

The proposed gait analysis system was tested considering several parameters. The objective of the implementation of the system is both to track and record the lower limbs' specific features as well as to provide an automatic classification tool of gait patterns. As such, we evaluated the quality of the extracted features and the classification's performance.

For this experiment, we had the collaboration of ten healthy volunteers, nine males and one female, between the ages of 25 and 35. The subjects were asked to walk in a straight line pushing the robotic walker. The following three gait patterns were used:

1) normal gait with no restrictions;
2) restriction of the right knee preventing the leg from bending, simulating a right knee joint disorder; and
3) restriction of the left knee preventing the leg from bending, simulating a left knee joint disorder.

Patterns 2 and 3 simulate an osteoarthritis knee condition, which presents an asymmetrical gait pattern on afflicted individuals. The subjects performed three repetitions of 50 consecutive steps for each pattern. Each repetition involved speed variations of 0.25 steps/s, 0.5 steps/s, and 1 step/s.

The first evaluation assessed the proposed system capacity to accurately track the lower limbs, which is an important feature for healthcare professionals to help them analyze how a patient is progressing during rehabilitation. Precision is crucial in gait analysis, as such, we compared some of our extracted features with data obtained by an XSens MVN inertial suit, which we considered as a ground truth. We evaluated the system's performance in the extraction of the hip's joint and knee's joint excursions for an asymptomatic gait pattern. Fig. 15(a) illustrates the performance of the system, providing a visual representation of the variation of each feature during a stride. Looking at the results, we can observe a consistent tracking by the proposed method when compared to the Xsens's system. We achieved an average error of 1.6°, validating this proposed method's reliability for tracking.

Furthermore, we also provide results for the same features as well as heel strike detection, during several strides of different gait patterns. Analyzing Fig. 15(b) and (c), we can observe a coherent tracking of each specific feature. For normal gait patterns, the motions of each leg vary symmetrically, whereas for a restricted knee pattern, the knee's joint variation is consistent with the leg's bending restriction, where one of the leg's knee joint angle remains constant during the gait cycles.

The classification strategy implemented in this system was evaluated using a leave-one-out cross-validation test. The purpose was to evaluate the generalization capacity of the classifier. The learning stage was performed with eight persons and tested against two unseen "new persons." Tests were performed with different kernel variations, linear and polynomial kernels. Table VI presents the results for each kernel. The cubic kernel stands out as the best classifier. The ROC curve shown in Fig. 16 evidences the discrimination capacity from the SVM with a cubic kernel.

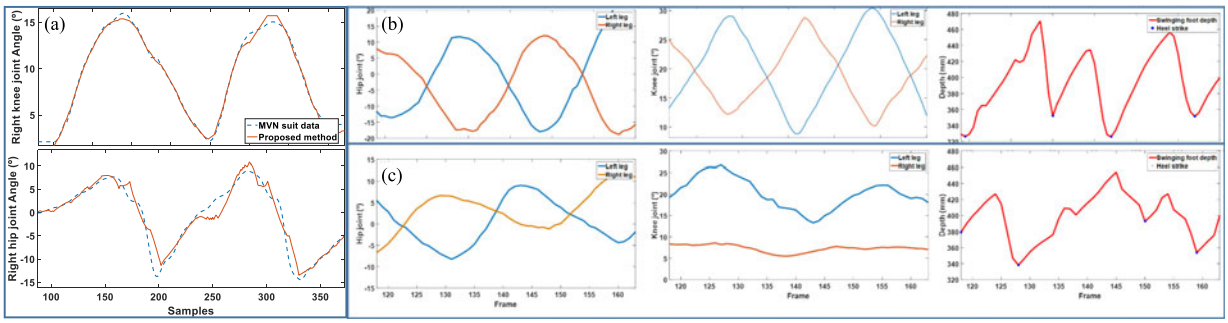

Fig. 15. Gait analysis system's validation (a) against ground truth, and tracking validation during three strides of the features: hip's joint angle, knee's joint angle, and swinging foot's depth with heel strike detection, seen in row (b) for a normal gait pattern and (c) restricted right knee described by an asymmetrical gait pattern.

TABLE VI
PERFORMANCE RESULTS OF THE SVM WITH DIFFERENT KERNELS

	Accuracy (%)	Precision (%)	Recall (%)
Linear	73.36	74.71	90.91
Quad.	86.14	88.99	90.44
Cubic	88.34	90.65	90.44

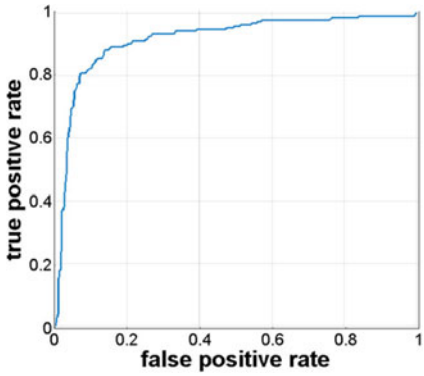

Fig. 16. ROC curve for the gait analysis system's classifier.

The system evidenced, however, some problems that are related to the technical specificities of the sensors used. The Leap Motion sensor relies on IR reflection from the scene; so, if the users were wearing dark and nonreflective cloths, the 3-D data retrieved from the sensor reveals itself unreliable. The same goes for all black footwear, where the structured light grid from the Intel's camera is not reflected. The fitting of the clothes also influences performance. Slimmer fitted clothes offer more reliable results. However, even with regular pants, the obtained results proved to be satisfactory. We consider these less favorable scenarios easily avoidable by providing a set of usage guidelines and as such, they do not hinder the advantages of the proposed system. Another limitation of the system is its inability to detect gait patterns that exhibit no separation of the legs at any point from the feet up to the waist at the contour image level. If the legs completely overlap, the performance of the system is affected. It is also important to note that our approach considers gait analysis in linear motion and discards rotations. Even using a 3-D approach, we have a limitation related to the turning motions of the user. If the user's body rotates and the frontal plane of the user is not considerably parallel to the walker, our approach may become unreliable, due to an incorrect sagittal plane projection.

We consider that the developed system exhibits a satisfactory performance, denoting an adequate feature extraction method that leads to an overall accuracy of the classifier for gait asymmetry pattern detection of 88.3%.

VII. CONCLUSION

This paper presented the development of a new robotic walker platform, emphasizing the contributions made in a new HMI paradigm and the introduction of gait analysis on board the walker. These innovations were supported by the use of low-cost technologies and a sensorial multimodal approach.

The HMI's vision-based paradigm changes the conventional force-sensing paradigm to one which allows an intuitive interaction and the enforcement of safety measures focused on a correct hand gripping for a robust body weight support.

Focusing on the rehabilitation potential of the walker as a therapy device, we proposed a gait analysis system, which relies on a multimodal sensor structure that is able to monitor the user in close proximity. The system perceives the lower limbs of the user, extracting specific features of gait, which are recorded and at the same time fed to a machine learning framework to classify gait asymmetries. This system could allow rehabilitation professionals to follow their patients' progress easily, relying on quantitative and qualitative data.

Experiments were performed with volunteers to assess each module of the walker's development. The tests of the HMI revealed that the subjects were comfortable with the operation of the walker and that the HMI accurately translated the navigational intentions of the subjects. On the gait analysis system's side, the tests revealed a satisfactory feature extraction and tracking along with a strong discrimination between normal gait and asymmetric gait.

We plan in the future to make the gait analysis system adaptable, tracking a user's gait pattern. The idea is to perceive if there is a shift in the gait pattern over time. This will provide a tool for healthcare professionals to identify gait pattern changes and plan adequate therapy sessions.

REFERENCES

[1] Y. Balash, M. Hadar-Frumer, T. Herman, C. Peretz, N. Giladi, and J. M. Hausdorff, "The effects of reducing fear of falling on locomotion in older adults with a higher level gait disorder," *J. Neural Transmiss. (Vienna, Austria, 1996)*, vol. 114, no. 10, pp. 1309–1314, 2007.

[2] H. Bateni and B. E. Maki, "Assistive devices for balance and mobility: Benefits, demands, and adverse consequences," *Arch. Phys. Med. Rehabil.*, vol. 86, no. 1, pp. 134–145, 2005.

[3] S. Dubowsky *et al.*, "PAMM—A robotic aid to the elderly for mobility assistance and monitoring: A 'helping-hand' for the elderly," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, pp. 570–576.

[4] G. Lacey, S. Mac Namara, and K. M. Dawson-Howe, "Personal adaptive mobility aid for the infirm and elderly blind," in *Assistive Technology and Artificial Intelligence*. New York, NY, USA: Springer, 1998, pp. 211–220.

[5] M. Alwan, A. Ledoux, G. Wasson, P. Sheth, and C. Huang, "Basic walker-assisted gait characteristics derived from forces and moments exerted on the walker's handles: Results on normal subjects," *Med. Eng. Phys.*, vol. 29, no. 3, pp. 380–389, 2007.

[6] O. Chuy, Y. Hirata, Z. Wang, and K. Kosuge, "Motion control algorithms for a new intelligent robotic walker in emulating ambulatory device function," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, 2005, pp. 1509–1514.

[7] A. Frizera, R. Ceres, J. L. Pons, A. Abellanas, and R. Raya, "The smart walkers as geriatric assistive device. The SIMBIOSIS purpose," *Gerontechnology*, vol. 7, no. 2, pp. 108–113, 2008.

[8] A. Frizera Neto, R. Ceres, E. Rocón de Lima, and J. L. Pons, "Empowering and assisting natural human mobility: The SIMBIOSIS walker," *Int. J. Adv. Robot. Syst.*, vol. 8, no. 3, pp. 34–50, 2011.

[9] A. Morris *et al.*, "A robotic walker that provides guidance," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, pp. 25–30.

[10] M. Patel, J. V. Miro, and G. Dissanayake, "A hierarchical hidden Markov model to support activities of daily living with an assistive robotic walker," in *Proc. IEEE Int. Conf. Biomed. Robot. Biomechatronics*, 2012, pp. 1071–1076.

[11] M.-F. Chang, W.-H. Mou, C.-K. Liao, and L.-C. Fu, "Design and implementation of an active robotic walker for Parkinson's patients," in *Proc. SICE Annu. Conf.*, 2012, pp. 2068–2073.

[12] M. Martins *et al.*, "Real time control of the ASBGo walker through a physical human-robot interface," *Measurement*, vol. 48, pp. 77–86, 2013.

[13] A. J. Rentschler, R. Simpson, R. A. Cooper, and M. L. Boninger, "Clinical evaluation of Guido robotic walker," *J. Rehabil. Res. Develop.*, vol. 45, no. 9, pp. 1281–1293, 2008.

[14] G. Lee, T. Ohnuma, N. Y. Chong, and S.-G. Lee, "Walking intent-based movement control for JAIST active robotic walker," *IEEE Trans. Syst. Man Cybern., Syst.*, vol. 44, no. 5, pp. 665–672, May 2014.

[15] C. A. Cifuentes, C. Rodriguez, A. Frizera-Neto, T. F. Bastos-Filho, and R. Carelli, "Multimodal human-robot interaction for walker-assisted gait," *IEEE Syst. J.*, vol. 10, no. 3, pp. 933–943, Sep. 2016.

[16] S. Fahn, "The freezing phenomenon in parkinsonism," *Adv. Neurol.*, vol. 67, pp. 53–63, 1995.

[17] M. Martins, C. P. Santos, A. Frizera-Neto, and R. Ceres, "Assistive mobility devices focusing on smart walkers: Classification and review," *Robot. Auton. Syst.*, vol. 60, no. 4, pp. 548–562, 2012.

[18] M. Martins, C. Santos, A. Frizera, and R. Ceres, "A review of the functionalities of smart walkers," *Med. Eng. Phys.*, vol. 37, no. 10, pp. 917–928, Oct. 2015.

[19] R. B. Davis, III *et al.*, "A gait analysis data collection and reduction technique," *Hum. Movement Sci.*, vol. 10, no. 5, pp. 575–587, 1991.

[20] K. Wan and H. Sawada, "3D motion prediction of human upper body by tracking reflective markers on a moving body," *Chest*, vol. 3, pp. T4 172–177, 2007.

[21] M. Gabel, R. Gilad-Bachrach, E. Renshaw, and A. Schuster, "Full body gait analysis with Kinect," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2012, pp. 1964–1967.

[22] A. M. Sabatini, C. Martelloni, S. Scapellato, and F. Cavallo, "Assessment of walking features from foot inertial sensing," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 3, pp. 486–494, Mar. 2005.

[23] C. Joly, C. Dune, P. Gorce, and P. Rives, "Feet and legs tracking using a smart rollator equipped with a Kinect," in *Proc. Workshop Assistance Serv. Robot. Hum. Environ. Conjonction IEEE/RSJ Int. Conf. Int. Rob. Sys.*, 2013.

[24] R. Z.-L. Hu, A. Hartfiel, J. Tung, A. Fakih, J. Hoey, and P. Poupart, "3D pose tracking of walker users' lower limb with a structured-light camera on a moving platform," in *Proc. IEEE Comp. Vis. Pattern Recognit. Workshop*, 2011, pp. 29–36.

[25] S. Page, M. M. Martins, L. Saint-Bauzel, C. P. Santos, and V. Pasqui, "Fast embedded feet pose estimation based on a depth camera for smart walker," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 4224–4229.

[26] J. Paulo, P. Peixoto, and U. Nunes, "A novel vision-based human-machine interface for a robotic walker framework," in *Proc. IEEE Int. Symp. Robot Hum. Interactive Commun.*, 2015, pp. 134–139.

[27] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965.

[28] J. Paulo and P. Peixoto, "Classification of reaching and gripping gestures for safety on walking aids," in *Proc. IEEE Int. Symp. Robot Hum. Interactive Commun.*, 2014, pp. 756–761.

[29] V. Vapnik, "Pattern recognition using generalized portrait method," *Autom. Remote Control*, vol. 24, pp. 774–780, 1963.

[30] J. Perry *et al.*, "Gait analysis: Normal and pathological function," *J. Pediatric Orthopaedics*, vol. 12, no. 6, pp. 815–821, 1992.

[31] T. Öberg, A. Karsznia, and K. Öberg, "Basic gait parameters: Reference data for normal subjects, 10–79 years of age," *J. Rehabil. Res. Develop.*, vol. 30, pp. 210–210, 1993.

[32] T. Oberg, A. Karsznia, and K. Oberg, "Joint angle parameters in gait: Reference data for normal subjects, 10–79 years of age," *J. Rehabil. Res. Develop.*, vol. 31, no. 3, pp. 199–213, 1994.

[33] E. H. Melis, R. Torres-Moreno, H. Barbeau, and E. D. Lemaire, "Analysis of assisted-gait characteristics in persons with incomplete spinal cord injury," *Spinal Cord*, vol. 37, no. 6, pp. 430–439, 1999.

[34] J.-Y. Bouguet, "Camera calibration toolbox for MATLAB," 2004. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/index.html

[35] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, no. 1–3, pp. 7–42, 2002.

[36] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, 2011, Art. no. 27.

João Paulo (S'14) received the M.Sc. degree in electrical and computer engineering from the University of Beira Interior, Covilhã, Portugal, in 2010. He is currently working toward the Ph.D. degree in robotic mobility aids and gait analysis in the Institute of Systems and Robotics, University of Coimbra, Coimbra, Portugal, under the supervision of Prof. Peixoto and Prof. Nunes.

His research interests include assistive and rehabilitation robotics particularly mobility assistances. He focuses on perception and machine learning for gait analysis and HRI.

Paulo Peixoto (M'09) received the Ph.D. degree in electrical engineering from the University of Coimbra in 2003.

He with currently an Assistant Professor at the Department of Electrical and Computer Engineering of the University of Coimbra and a researcher at the Institute of Systems and Robotics, University of Coimbra (ISR-Coimbra). He has been involved/responsible for several funded projects at both national and international levels, in the areas of computer vision, assistive robotics, and intelligent vehicles.

His research interests include computer vision applied to intelligent transportation systems, pattern recognition, and human-computer interaction.

Urbano J. Nunes (SM'09) received the Ph.D. degree in electrical engineering from the University of Coimbra, Coimbra, Portugal, in 1995.

He is currently a Full Professor with the Electrical and Computer Engineering Department, University of Coimbra. He is the coordinator of the Automation and Robotics for Human Life Group of the Institute for Systems and Robotics. He was involved with/responsible for several funded projects at both national and international levels in the areas of human-centered mobile robotics and intelligent vehicles.

Prof. Nunes is an Associate Editor of the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS. He was a General Chair of the 2012 IEEE International Conference on Intelligent Robots and Systems and 2017 IEEE International Symposium on Robot and Human Interactive Communication.