2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

Development of Foot Displacement Detection Algorithm for Power Wheelchair Footplate Pressure and Positioning

Steve J. A. Majerus[1], *Senior Member, IEEE,* Jeremiah Ukwela[2], Joseph Lerchbacker[1], Kath M. Bogie[1,3], and M. Kristi Henzel[1,2,4]

***Abstract*— Inadvertent lower extremity displacement (ILED) puts the feet of power wheelchair (PWC) users at great risk of traumatic injury. Because disabled individuals may not be aware of a mis-positioned foot, a real-time system for notification can reduce the risk of injury. To test this concept, we developed a prototype system called FootSafe, capable of real-time detection and classification of foot position. The FootSafe system used an array of force-sensing resistors to monitor foot pressures on the PWC footplate. Data were transmitted via Bluetooth to an iOS app which ran a classifier algorithm to notify the user of ILED. In a pilot trial, FootSafe was tested with seven participants seated in a PWC. Data collected from this trial were used to test the accuracy of classification algorithms. A custom figure of merit (FOM) was created to balance the risk of missed positive and false positive. While a machine-learning algorithm (K nearest neighbors, FOM=0.78) outperformed simpler methods, the simplest algorithm, mean footplate pressure, performed similarly (FOM=0.62). In a real-time classification task, these results suggest that foot position can be estimated using relatively few force sensors and simple algorithms running on mobile hardware.**

***Clinical Relevance*— Foot collisions or dragging are severe or life-threatening injuries for people with spinal cord injuries. The FootSafe sensor, iOS app, and classifier algorithm can warn the user of a mis-positioned foot to reduce the incidence of injury.**

I. Introduction

Power wheelchairs (PWC) are critically important for independent mobility and quality of life for persons with spinal cord injuries and disorders (SCI/D), but even optimally configured PWC's can be dangerous during operation by users who cannot feel, see or easily reposition their lower limbs when driving conditions, e.g. rough terrain, cause inadvertent lower extremity displacement (ILED) from the PWC footplates. Additionally, pressure relieving features of PWC such as tilt-in-space with leg elevation may displace the feet because the footrests must extend away from the feet during leg elevation. After ILED from the footplates, traumatic fractures, wounds/cuts/ infections, abrasions, pressure injuries, and/or amputations may occur [1]. Types of ILEDs reported have included the foot slipping off the footplate without the user being aware, catching the feet between the footplate and an object, bumping the feet into objects, the feet hanging over the footplates, the foot dragging under the PWC, or developing pressure "hot spots" causing skin breakdown [2]. Medical costs of lower limb injuries associated with foot mispositioning on the PWC footplate are known to be high [3], despite the relative ease of repositioning the foot if the user is warned of ILED.

Active safety systems are now common in automobiles to improve automobile control and prevent crashes [4], and in mobile robots, such as pharmacy delivery systems in hospitals [5]. "Smart" wheelchairs with sensors to actively detect the surrounding environment are a new commercial reality [6], however there are no current systems to help users replace the missing sensory feedback of the nervous system which is lost after SCI/D. Safe wheelchair use relies upon users consistently implementing safe operation techniques, which requires intact vision, cognition and impulse control [7]. Development of smart footplate position sensing and feedback provides the basis for developing active safety interventions to address the unmet clinical need for real-time prevention of lower limb injuries during PWC use [8], first by helping define incidence of lower limb injuries and near misses and, once optimized, by helping to prevent ILED-related injuries.

The FootSafe system (Fig. 1) combines force and proximity sensors with a wirelessly-connected smartphone application with adjustable alarm options to provide continuous real-time monitoring of foot position on the footplate. The device shown was designed to fit the a commonly used PWC footplate. Each half of the FootSafe system includes an array of 24 low profile piezoresistive force sensing regions (FSR) and 14 infrared (IR) proximity sensors to detect foot position above the footplate. In this work, we describe the first efforts to use the Footsafe system with PWC users, and the performance of several detection algorithms at detecting ILED in real time.

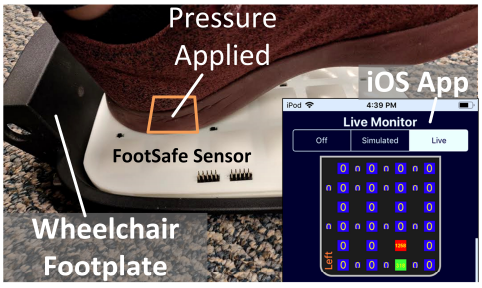

Fig. 1. The FootSafe sensor transmits foot pressure and position data from the footplates of a wheelchair. Data from 24 locations per footplate are analyzed by an iOS application in real time to warn the user of inadvertent lower extremity displacement.

II. FootSafe Real-Time Sensor Implementation

The FootSafe sensor was designed to accommodate a wide range of footwear, be cost-effective, and unaffected by environmental conditions or weather. FootSafe used sensors to detect foot pressure profiles across the footplates of a standard PWC. The system incorporated force-sensing resistors (FSRs) and infrared (IR) distance sensors to determine the foot's position based on pressure and proximity to the footplate. Details on sensor architecture were previously described [8] and are only briefly summarized here. Further, while the

FootSafe includes pressure- and distance-based sensors, here, we will only focus on the pressure sensor aspects because these were used to train detection algorithms described later. Distance-based algorithms are a future topic.

A. *Footplate sensor electronics*

FSRs were implemented using interdigitated electrodes on a printed circuit board (PCB), with a conductive polymer (XactFSR film, Sensitronics, Bow, WA) affixed above each electrode site [8]. Each of the 24 FSRs per footplate measured 2.8 x 2.8 cm. FSRs were read out with a row-column multiplexed scheme allowing each FSR to be individually measured as a resistive half-bridge. A Teensy 3.6 (ARM Cortex M4, 180 MHz, PJRC, LLC) microcontroller controlled the multiplexers and used an onboard analog-to-digital converter (ADC) to measure FSR values at 10 Hz. Sensor values were truncated to 10-bit accuracy to produce a relative mass sensitivity of about 1 g per FSR.

FSR data recording and pre-processing was performed on the microcontroller to produce a new sensor frame every 100 ms. Sensor frames from both feet were measured, processed, and transmitted using a serial Bluetooth (HM-10) interface.

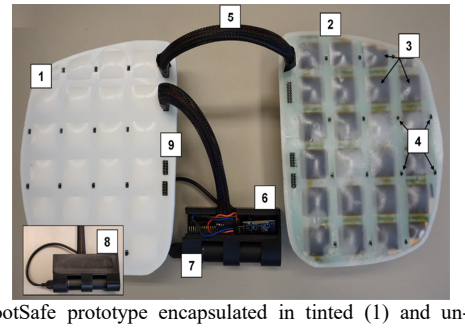

Fig. 1. FootSafe prototype encapsulated in tinted (1) and un-tinted (2) urethane to show FSRs (3) and IR sensors (4). An interconnect cable (5) joined the left and right sensors to an enclosure (6-8) housing the Teensy 3.6, HM-10 Bluetooth module, and USB battery.

B. *Real-time data readout and logging app*

Sensor data transmitted from the FootSafe was processed by a mobile device (Apple iPod Touch) with a custom iOS application (Fig. 2). The application applied calibration data to raw sensor values, recorded data for analysis, and alerted the user of ILED based on a simple algorithm previously described [8]; later we describe more accurate real-time algorithms tested with data from human participants.

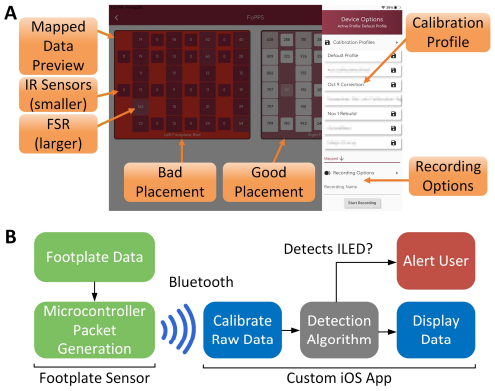

Fig. 2. An iOS app (a) received sensor data over Bluetooth and alerted the PWC user if a foot was poorly placed (b).

C. *Footplate sensor array construction*

FootSafe electronics were assembled on fiberglass printed circuit board with an electroless nickel, immersion gold plating. After functional testing, circuit boards were overcoated to prevent damage during foot contact and PWC operation. A two-piece protective overcoating of Simpact™ 85A polyurethane (Smooth-On, East Texas, PA) was cast in aluminum molds. Small 3x3 cm squares of 2 mm thick XactFSR film (Sensitronics, Bow, WA) were attached to the top layer of the PCB using double-sided tape (McMaster-Carr, Elmherst, IL) such that they rested directly above each FSR (Fig. 3). A 2-mm tall bump in the top of the polyurethane overcoating was positioned above each FSR on the FootSafe to act as a stress concentrator, while a small cutout window rested above each IR sensor. The IR sensor window could be filled with IR-transparent epoxy, but this was not done for the tested prototypes. The bumps ensured that pressure from a patient's foot would be distributed onto an FSR, and not another part of the FootSafe PCB. A silicone adhesive (Dow Corning 3145) glued the two polyurethane parts together, fitting over the PCB and completely encapsulating it.

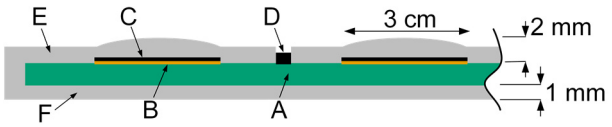

Fig. 3: Force sensors were integrated on a PCB (A) with gold-plated electrodes (B) contacting conductive film (C). Data from infrared distance sensors (D) were not analyzed in this work. The entire assembly was encapsulated with flexible urethane (E,F) for environmental resistance.

D. *Pressure sensor array calibration*

When biased at a fixed voltage of 3.3 V, the nonlinear sensor voltage-pressure profile of the FSRs was modeled as

$$V = 3.3\left(1 - e^{\frac{-F}{K}}\right),$$

where F was the equivalent force applied, and K was a unit conversion factor. To simplify calibration, we adopted a linear fit to force in a quasi-linear range (Fig. 4).

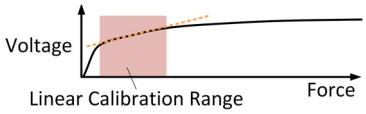

Fig. 4. FSRs demonstrated a force response; to simplify calibration a linear fit to a limited range of the response curve was used.

To calibrate each sensor, linear regression was used to fit an offset and slope, using predetermined calibration masses of 0, 500, and 1,000 g. These masses were chosen based on the leg mass [9] of a 6-foot man with a BMI of 30, and the geometry of the footplate surface. 48 slope and intercept pairs were calculated, one for each force site. This information was saved in a custom data structure. During real-time use, offset and slope calibration for each reading was extracted from the data structure to produce a floating-point number in units of grams.

III. Algorithmic Detection of ILEDs

Classification algorithms detect when the foot is in an unsafe position and alert the user. To understand the natural variance in foot positions during common activities, data were collected from humans seated in a PWC. The foot was placed in normal and unsafe positions to create a dataset. Six

algorithms were tested on this dataset to determine how accurately unsafe foot position could be detected in real time.

A. Human data collection study overview

Research was performed with approval of the Cleveland VA Medical Center Institutional Review Board. Seven individuals, one of whom was an experienced PWC user with SCI/D participated (3 women, 4 men). Participants were of varying heights (157-183 cm) and shoe sizes, and had varying fits in the PWC (oversized to undersized). All recordings were performed using a Permobil F5 PWC and the FootSafe device affixed to the footplate tray using 3M VHB tape (Fig. 5). One recording of simulated daily PWC use was made for each study participant, and three 10-minute recordings at rest were made for the experienced PWC user. The experienced PWC user with SCI/D had several sets of simulated daily PWC use collected to account for potential bias. Each participant performed a set of movements known to provoke ILED:

1. The ball of each foot was pivoted upwards from the heel
2. Each foot everted outwards from the footplate
3. Each foot was removed and then returned to the footplate
4. The PWC was tilted backwards to approximately 45°
5. The backrest of the PWC was fully reclined backwards
6. The legrest of the PWC was fully extended
7. The backrest of the PWC was brought fully upwards
8. The PWC was tilted forwards to 0°
9. The PWC legrest was returned to neutral sitting position

Fig. 5. Example seating positions in PWC used in human data collection.

B. Example pressure data at specific foot locations

The FootSafe system performed real-time classification of foot position based on frames of sensor data captured at 10 Hz (Fig. 6). Although only 24 FSRs per foot were used to determine footplate pressures, because the foot is large and continuous, this resolution was sufficient to show gross differences in pressure distribution based on foot position. All frames were independently classified in this analysis, although sliding window or majority vote approaches could be used in the future to reduce single-frame classification errors.

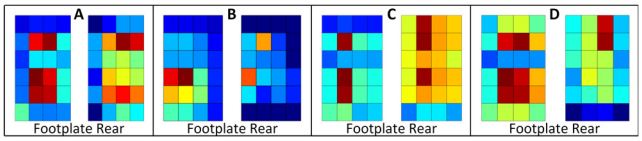

Fig. 6. Example FSR values with (a) both feet in safe positions, (b) ILED on both feet, (c) ILED on left foot, and (d) ILED on right foot. The color scale range was adjusted for each example for side-by-side comparison.

C. Algorithm development and optimization

To test each foot position classification algorithm, we defined a figure of merit (FOM) to emphasize the importance for true positive detection (Table 1). The training data set was manually divided into normal and misplaced foot positions.

TABLE I. TERMS USED IN ALGORITHM FIGURE OF MERIT

Term	Name	Description
α	True Negative	Correctly detected as misplaced
β	True Positive	Correctly detected as correctly placed
K	False Positive	Incorrectly detected as correctly placed
γ	False Negative	Incorrectly detected as incorrectly placed
L	Frame Length	Number of frames in recording
M	Normalization Constant	Maximum value the fitness function produces for a completely accurate detection

The FOM, σ, with range [-1 1] was calculated as:

$$\sigma = \frac{\frac{\alpha - K}{L} - \left(\frac{\gamma}{L}\right)^2 + \left(\frac{\beta}{L}\right)^3}{M}$$

and the normalization constant M was calculated as:

$$M = \frac{^{\alpha}max}{L} + \left(\frac{^{\beta}max}{L}\right)^3.$$

Six algorithms were evaluated with the human dataset. Five algorithms were developed heuristically based on *a priori* knowledge of safe foot position. These algorithms were also developed to be easy to compute in a real-time scenario on low-power hardware. All heuristic algorithms were binary, thresholded classifiers; thresholds were chosen for each algorithm to maximize the FOM. A cubic k-nearest neighbors (**KNN**) algorithm was also studied to evaluate the potential accuracy improvement in using a machine learning approach.

The first algorithm (**Mean**) calculated the mean force on the footplate, using all sensor channels. This calculated the average equivalent mass that the lower limbs were applying to each footplate surface. This value was compared to a set threshold, and the frame was marked as misplaced if both the calculated average mass was below the threshold. The threshold value was chosen as approximately 20% of the calibration mass, and optimized to increase the FOM.

The second algorithm (**Centroid**) calculated the location of the centroid of equivalent mass applied to the footplate surface. A region of interest was defined as a maximum allowable distance for the centroid from the center of the footplate. The average mass on the center two rows of FSRs were also compared to a threshold. The frame was marked as misplaced if the centroid was outside of the region of interest and the average mass was below threshold (Fig. 7a).

For the third, fourth, and fifth algorithms, the FSR matrix was divided into three regions (Fig. 7b). The third algorithm (**Relative**) calculated the average mass per region. The difference of the average of Region 1 and Region 3 was compared to a threshold, while the average of Region 2 was compared to a separate threshold. If the Region 3 – Region 1 difference was above threshold, and the average equivalent mass in Region 2 was below threshold, the frame was marked as misplaced. This algorithm was intended to detect improper heel or metatarsal joint placement. For all tests, the difference threshold was set to 50 g which provided rejection of noise while being sensitive enough to detect foot pressure.

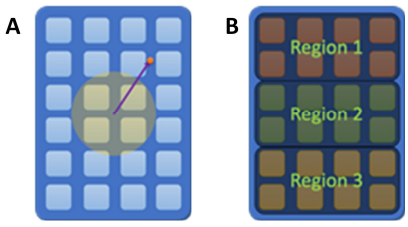

Fig. 7. A centroid algorithm (A) was evaluated in addition to algorithms which compared differences between FSR regions on the footplate (B).

The fourth algorithm (**Ratio**) was similar to **Relative** but instead a ratio between Region 1 and Region 3 was calculated. A frame was marked as valid when the ratio of masses fell within a specified range. This algorithm was designed to detect an uneven distribution of pressure.

The fifth algorithm (Mean Square Error, **MSE**) calculated the mean squared error of the equivalent applied mass of Region 1 and Region 3 as compared to the threshold value of Region 2. The calculated value was compared to a threshold value, as well as the equivalent mass of Region 2.

The **KNN** algorithm was trained using the MATLAB Classification Learner (R2020a) with each calibrated force sensor treated as a feature. A misclassification matrix was used to train the model (Table 2). Model performance was validated using three-fold cross-validation on all available frames from all experiment trials (n=41,056). Cross validation prevents overfitting by partitioning the provided dataset and estimating accuracy on each fold. Every recorded sensor frame was manually marked with the state definition (None, Left, Right, or Both). Each fold contained 1/3 of the data from the study, randomly selected but constrained to contain an equal number of observations per subject. The KNN essentially considered the correlation of a given footplate state to known states within its training set. Fig. 8 shows an example parameterized correlation of two adjacent force sensors. Across these two sensors, an equivalent applied mass of 750 g on Sensor 1 and 600 g on Sensor 2 would be estimated as a right ILED. Approximately 2,300 of these parametric maps were generated, each with their own classification weights.

TABLE II. MISCLASSIFICATION MATRIX FOR KNN TRAINING

		Predicted Class			
		None	Left	Right	Both
True Class	None	0	2.5	2.5	2.5
	Left	2.5	0	5	1
	Right	2.5	5	0	1
	Both	2.5	1	1	0

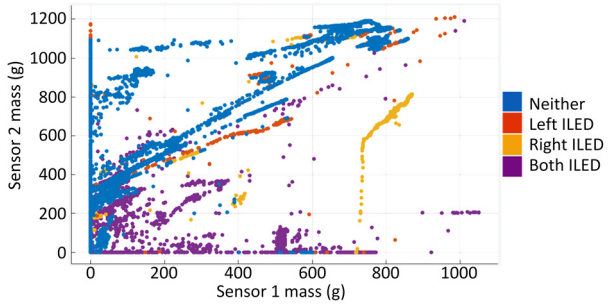

Fig. 8. Example KNN parameterized correlation of two adjacent sensors.

D. Algorithm performance

The mean FOMs, accuracies, and positive detection rates (PDR) for each algorithm were calculated (Table 3). The PDR described the percentage of ILED properly detected by each algorithm, as opposed to the percentage of all frames properly detected that the accuracy represents. The standard errors for each statistic were also calculated.

The KNN model consistently outperformed the other algorithms, however, the KNN model would be significantly more complex to implement in a real-time classification task due to the large number of features and calculations required. Of the simpler algorithms, the Mean and MSE approaches showed reasonable performance.

TABLE III. AVERAGE ALGORITHM PERFORMANCE MEASURES

	Mean	**Centroid**	**Relative**
FOM	0.62±0.10	0.25±0.16	0.20±0.13
Accuracy	84.2±4.1%	68.8±7.4%	67.2±6.3%
PDR	77.5±6.8%	29.5±10.5%	48.8±7.8%
	Ratio	**MSE**	**KNN**
FOM	0.29±0.12	0.37±0.10	0.78±0.08
Accuracy	70.7±5.0%	72.9±4.5%	95.3±2.7%
PDR	51.6±8.1%	63.0±8.2%	90.5±3.9%

The algorithms which relied on information from Region 2 of the force matrix tended to score lower. This may be due to the geometry of the footplate surface and the user's footwear. These algorithms worked under the assumption that the user's foot makes full contact with all regions of the footplate surface, which may not be the case depending on footwear. For example, when the arch of the user's shoe spanned Region 2 of the footplate, as prevalent with athletic shoes and shoes with heels, the center portion did not make contact. When the threshold value for Region 2 was decreased for these algorithms, an improvement in determining ILED correctly was obtained.

The Centroid algorithm had the lowest FOM, but also had the highest standard error, indicating a higher variance in the recorded values. Even considering the standard error, the PDR of the Centroid algorithm was the lowest out of all tested algorithms. It's possible this may be attributed to footwear geometry, or the Centroid calculations may be inaccurate due to the low resolution of the force sensing matrix.

As compared to the manually programmed algorithms which only required a few cycles of computation time, the machine learning model required approximately 40-60 ms for classification on a modern mobile device (Apple iPod Touch). When this level of computational power is available, machine learning algorithms like KNN are feasible in a real-time application. Feature reduction could potentially reduce the computational burden of the ML algorithm.

REFERENCES

[1] P. Paparone, "Lower extremity ulceration caused by medical scooter injury: A case series," *Geriatric Nursing*, 34(1), 2013.

[2] W. Y. Chen, Y. Jang, J. der Wang, *et al.*, "Wheelchair-related accidents: relationship with wheelchair-using behavior in active community wheelchair users," *Archives of Physical Medicine and Rehabilitation*, 92(6), 2011.

[3] L. R. Morse, R. A. Battaglino, K. L. Stolzmann, *et al.*, "Osteoporotic fractures and hospitalization risk in chronic spinal cord injury," *Osteoporosis International*, 20(3), 2009.

[4] T. Liu, Selpi, R. Fu, "The Relationship between Different Safety Indicators in Car-following Situations," in *IEEE Intelligent Vehicles Symposium, Proceedings*, 2018.

[5] M. R. Summerfield, F. J. Seagull, N. Vaidya, *et al.*, "Use of pharmacy delivery robots in intensive care units," *American Journal of Health-System Pharmacy*, 68(1). 2011.

[6] K. Schilling, H. Roth, R. Lieb, *et al.*, "Sensor Supported Driving Aids for Disabled Wheelchair Users," *IFAC Proceedings Volumes*, 31(2), 1998.

[7] R. C. Simpson, E. F. LoPresti, R. A. Cooper, "How many people would benefit from a smart wheelchair?" *Journal of Rehabilitation Research and Development*, 45(1), 2008.

[8] S. J. A. Majerus, J. Lerchbacker, D. Barbaro, *et al.*, "Power Wheelchair Footplate Pressure and Positioning Sensor," in *IEEE Eng. in Med. and Biology Soc, Proceedings*, 2018, pp. 4367–4370.

[9] S. Plagenhoef, F. G. Evans, T. Abdelnour, "Anatomical Data for Analyzing Human Motion," *Research Quarterly for Exercise and Sport*, 54(2), pp. 169–178, 1983.