EVALUATION OF PATTERN RECOGNITION METHODS FOR ASCENDING STAIR DETECTION FROM LIDAR FOR WEARABLE REHABILITATION DEVICES

Christian Mele, 20605285 *
Department of Systems Engineering
University of Waterloo
Waterloo, ON, Canada
cmele@uwaterloo.ca

Shovon Saha, 21042448
Department of Mechanical and Mechatronics Engineering
University of Waterloo
Waterloo, ON, Canada
ss2saha@uwaterloo.ca

ABSTRACT

Stair detection and characterisation is critical for the use of active wearable assistive lower limb devices in many common environments. Current state-of-the-art methods rely heavily on model based approaches that are constantly searching scenes to generate virtual stairs. The computational effort required limits their usability. In this study, a range of model-free pattern recognition based methods for identifying stairs is presented with commonly used LiDAR preprocessing methods applied to assess which function best. Simple logistic-regression and support vector machine algorithms proved to function at a relatively high accuracy, 83% indicating with further implementation model-free stair identification algorithms can be used to inform model-based stair classification.

1 INTRODUCTION

Lower-limb wearable rehabilitation devices (exoskeletons, orthoses, prostheses) are active and passive devices worn by users with some form of mobility impairment to provide assistance and support for gait and stability Huo et al. (2014). Active lower-limb devices deliver a combination of preset and online adjusted assistance profiles to the user to aid, or replace, gait trajectories. Altering assistance profiles to avoid or traverse terrain and topology alternative to traditional flat ground, requires extensive knowledge on the surrounding environment and is a constant challenge in robotics and control. Specifically, for wearable robotics, stair identification and navigation is a complex issue when determining efficient and effective methods for identification and traversal Kurbis et al. (2022).

Stairs offer a complex problem in assistance, relying on knowledge of the scene and surfaces with the explicit intent to ascend or descend precarious terrain. With the growing popularity of assistive devices, increased use cases introduce the potential for increased interaction with stairs for traversal in unmapped environments. Accurately identifying, fully evaluating and characterising stairs is critical to the safety of the user of a wearable assistive device for traversal. The focus of this paper is centred around identification of stairs as obstacles for traversal from sensors onboard the device.

Stair detection and characterisation is an extensively investigated and researched topic within robotics and wearable devices **?**. State of the art sensor platforms (LiDAR, RGBd, Depth Sensing etc.) combined with highly advanced model based identification offer a constant and robust identification method for fully characterising stairs. Many of these model based solutions require constant streaming and evaluation of scene data, whether or not stairs are in the environment.

There is a need for a method of sensing and identifying stairs and stair-like structures, beginning with ascending stairs, that does not rely on highly complex model based sensing methods and highly accurate sensor platforms. By developing and investigating methods for stair identification prior to complex model fitting, it may be possible to employ much less efficacious sensing platforms for similar results without the computational, and monetary, cost.

*For additional questions or details about this paper submission in completion of Syde 675, please email cmele@uwaterloo.ca

1.1 Background

Stair detection algorithms generally focus on a model based approach with additional machine learning algorithms used to classify and characterise stairs Wang & Zhang (2019). Groundwork already established in literature may be used as inspiration for developing and evaluating novel methods for stair detection as opposed to the full model based approach. This section highlights the benefits and drawbacks of the current state of the art for stair detection and characterisation to provide inspiration and direction for an ascending stair detection only classifier.

The majority of stair characterization algorithms operate using a model based framework that is constantly operating to search for physical stair-like features in an observable scene Harms et al. (2015). The most common physical properties searched for in stair detection relate to the planar supporting surfaces, rising stair surfaces and vertices (or lines at the edge of stairs). Physical assumptions are often placed on these features that help sort out and generate reference stair structures in 3D space: e.g. width, depth, minimum heights, connections, etc. Sriganesh et al. (2023). This method, while based on physical properties and definitions, is limited by assumptions in identification, and computational time in running. When stair interfaces are not present these algorithms still search for planes and potential vertices as components of stairs - implementing large transforms and logistical graphs to search for stair geometry.

To support the model based framework, the use of pattern recognition methods have been implemented to define what kind of stairs are being observed after being fully characterised. Support Vector Machines (SVMs) use model specific features as a vector and observe whether or not ascending, descending, or no stairs are being observed from sensor data. When RGB-D cameras are employed, convolutional neural nets (CNNs) function as a model free approximator of stair observation to classify and match the model estimation.

Current methods for stair detection rely heavily on high cost and highly accurate sensor platforms, with the assumption that stairs are within or near view of the sensor platform Kim et al. (2024). This poses an issue, however, in cases where computational capabilities and sensor quality is worse. Many of the physical model constructors would be difficult to implement, or highly inaccurate and slow when working with much worse sensors that drop information or operate under much smaller fields of view.

Stair detection and characterisation methods have been heavily investigated and operate effectively under robust testing conditions and with highly accurate sensor platforms. This level of computation is not necessary, nor warranted in cases where stair detection is also not necessary at all times during device use. The only time at which stair characterisation should be performed is when stairs are present in scene, and a lower computational-effort detection method would allow for more complex model based methods to activated and run, even in scenarios where sensor quality is worse.

1.2 Objective

The objective of this paper is to evaluate and implement machine learning based stair detection methods, as opposed to model based, to detect when an individual wearing a LiDAR sensing platform is facing a set of stairs. By investigating common and simple binary classifiers in machine learning, and using simple feature detection methods from LiDAR data, it is hypothesised that a lower-fidelity sensing platform will be able to classify whether a scene contains ascending stairs. This information may then be used as an indicator to evaluate stair characteristics with much more complex model based evaluation methods, without the same level of computational cost. Fundamentally, it is shifting the stair classification methods applied at the end of most physical models to inform which type of stair is in the scene after modelling, to prior utilising model-free methods to save effort and time.

To achieve this, a low-fidelity LiDAR collection rig is set up and a preliminary dataset of ascending stairs, near stair and no-stair scenes is collected from a range of distances. Using this data, simple feature detection methods are applied (PCA analysis) that do not rely on physical model properties, which is then processed through three separate machine learning algorithms. The three algorithms implemented, a state vector machine (SVM), logistic regression model and random forest represent three common but separate approaches to binary classification in machine learning. The resulting effectiveness in classifying ascending stair detection from training and testing data is then analysed as a potential solution for stair identification prior to the need of physical model characterisation.

2 Methods

To collect scene information a LiDAR collection apparatus was designed and fabricated to be worn by a testing participant for various scene information. The LiDAR selected for use (Cygbot, Cyglidar D1) has a 2-metre range and 120 degree field of view. LiDAR systems store no photo data and thus comply with HIPAA on patient anonymity. The device is mounted with a phone for validation onto a custom 3D-printed chest mount for collection, which can be seen worn in Figure 1. This sensor mount is connected to a NVIDIA Jetson Nano microprocessor running a ROS process to collect and store LiDAR data.

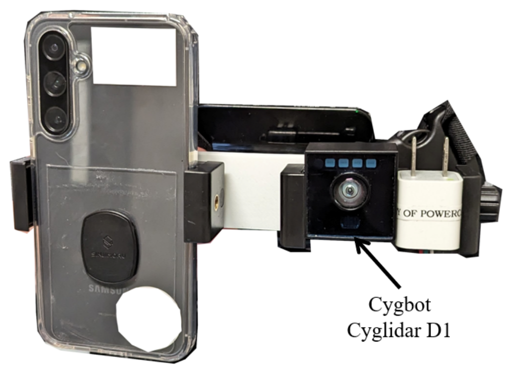

Figure 1: Cygbot Cyglidar D1 Collection mount design with accompanying phone for validation

2.1 Defining Stair Detection Threshold

Only ascending stairs are considered for this preliminary investigation, however adjustments made to training data should allow for identification of other stair structures. In order for stair identification to be successful, conditions for when it is ideal for stairs to be identified must be outlined. These identification parameters inform when a set of collected LiDAR data should be labelled as ascending stairs and is the desired result from training.

The intent to commence or observe stair traversal is split into angle of approach and range of approach to the stairs. Beginning with angle of approach it is assumed that an individual who is walking by a set of stairs has no intent for traversal. Stair approach angles are defined as 45 degree arcs stemming from the point perpendicular to the first stair. For this study, LiDAR data is trained as ascending stairs when viewed within the arc. Figure 2 visualises this collection arc.

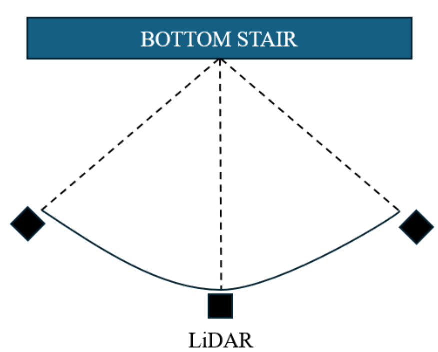

Figure 2: Stair Detection Threshold Arc used to define collection labelling

2.2 Collection

Collection was performed over a range of stair, near-stair and not-stair scenes. A number of ascending stair-wells were selected for examination within various buildings on the University of Waterloo Engineering campus. To get a better idea of variation the range of stair training, a set of wooden stairs used for humanoid robotics testing in Figure 3 (left), and a set of stairs present near a lecture hall split by a concrete column (Figure 3, 4th image). Each set of stairs was collected with a variation of angles and distances within the desired identification range shown in Figure 2. Different stair depths, sizes, count etc. were collected with obstructions to add to training robustness.

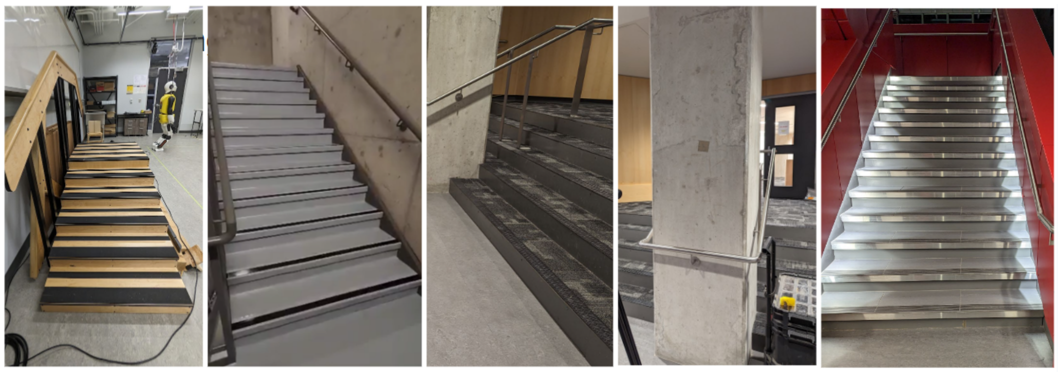

Figure 3: Example set of stair scenes collected to represent the positive condition to start running more complex model based stair characterisation algorithms. This includes highly occluded stairs.

Near-stair and not stair scenes are collected from a wide variety of random scenes around the same environments stair data is collected from. Near-stair scenes primarily include stairs observed from outside the allowable range and arcs so as to more accurately define when stairs should be identified. Non-stair scenes include walls, empty rooms etc. in an attempt to simulate scenes that may be seen during regular use.

30 second collection intervals were selected, to guarantee a variance in collected information from natural movements and adjustments (breathing etc.). 30 second intervals were collected to allow for pattern recognition algorithms to characterise noise present in scene data. During training each frame is analysed, so additional frames provide more robust training to noise. An example of collected scenes can be seen in Figure 4.

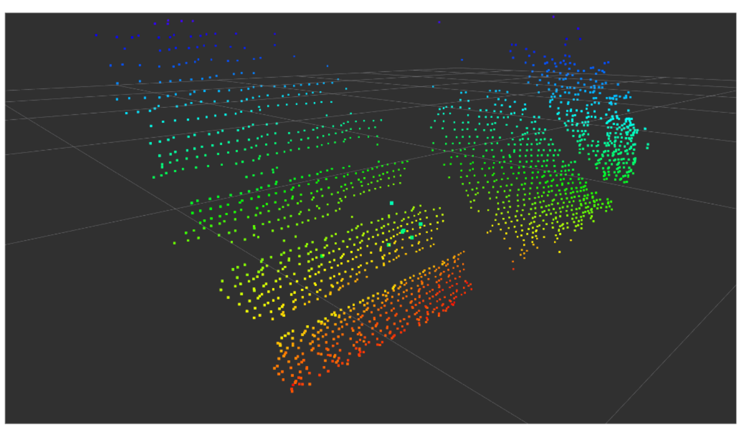

Figure 4: Example collection scene in Rviz displaying LiDAR scenes for a metal stairwell.

In total 8380 images were collected, sorted and labelled containing stair and non-stair data. 20% of these images were set aside for testing evaluation. Table 1 contains the training count information.

Table 1: Testing and Training LiDAR Frame Count

Training Image Count	6704
Testing Image Count	1676

2.3 PREPROCESSING

Preprocessing is a necessary step in LiDAR data collection. Raw LiDAR data is potentially too large for online evaluation for timing during typical use and thus needs to be transformed in some way to reduce the complexity of data, and bring out important information for feature detection.

Many alternative preprocessing methods exist for LiDAR and RGB-D data that can be used to reduce complexity of data and isolate important features. For the purpose of this study three different forms of preprocessing are investigated. Each are highlighted in Table 2, with a short description of their effect on the data, as well as a reminder of what data without preprocessing represents. This is to find what steps may be taken to speed up accuracy prior to more complex algorithm implementation.

Table 2: Preprocessing Method and Description of general approach

Preprocessing Method	Description
None - RAW data	Raw data consists of a scene's point data in the form of Distance, Angle and Height from the LiDAR translated into X,Y,Z frame data
Window Cropping	Window cropping removes data at the edges of the LiDAR frame - usually the noisiest region of scene collection
Depth Cropping	Depth cropping removes data near into the frame isolating components further back under the estimation that important information will be further into the scene
Voxel Based Downsampling	Voxel based downsampling fits voxel arrays over a set of data at each frame and then reduces the number of points within the voxel to a singular point. This method is common within LiDAR and RGB-D scene preprocessing [refs]

2.4 FEATURE DETECTION

Feature detection is informed and inspired by methods undertaken in literature for more complex characterisation methods, simplified for faster processing and evaluation. Most feature extraction methods for stair detection, as mentioned prior, focus on physical properties. Planes and vertices, and their respective properties, are often the most common features to extract from scene information as it may be used to verify stair structure and may be used as inspiration for determining simpler features to implement.

One such component for feature detection is principal component analysis (PCA). Principal component analysis pulls the dominant eigenvalues and eigenvectors out of a set of data by singular value decomposition or eigendecomposition. The nature of PCA directly describes the dominant covariances within a set of data, and for the purposes of 3D stair analysis may be able to elucidate the variance within stair LiDAR data in terms of planar normals as a physical correlate.

PCA, therefore, can be used to break any set of LiDAR data, regardless of sample size seen by the LiDAR from occlusion or downsampling, into a set of information that is broken into 3 major axes. These eigenvalues and eigenvector patterns for a range of ideal stair structures in 3D space are used as the features for training, as they are likely to be uniquely distinct containing alternating patterns of surfaces in point cloud data with variations that are not likely to exist in other non-stair scenes. For this reason, the only feature extraction performed in this study is PCA n=3 component analysis was performed on the preprocessed data.

2.5 Pattern Recognition Algorithms

The primary goal of this study is a binary identification of ascending stairs in LiDAR data. Three different classifiers have been selected for investigation that operate in distinctly separate ways to generate a decision boundary splitting stair and non-stair scenes.

The input for each of these models is a stack of the PCA information. A column vector of 11 variables is supplied in the form of:

$$[eigenvalue_1; eigenvector_1; .....eigenvalue_3; eigenvector_3]$$

The first classification model implemented is a Support Vector Machine (SVM) with a decision boundary based on Minimum Euclidean Distance (MED). The training set of data (containing stair, and no-stair data), uses the means of each labelled group as the prototype. This argument is evaluated, with the minimum value defining classification taken in the form of:

$$dist = \sqrt{\sum_{i}^{N}(x_{test_i} - x_{prototype})^2}$$

The second classification model implemented is a Logistic Regression Model (LRM). An LRM is a singular perceptron or neuron of a neural net layer intended for binary classification. A series of N weights (W_N) and a bias term (W_0) are trained using the training data to minimise the negative log-likelihood probability loss through gradient descent. The logistic relationship below is trained to minimize classification inaccuracy using testing data over a number of Epochs (1000 specifically), and then evaluated for accuracy on testing stair data.

$$\begin{Bmatrix} 0 & if & g(wx + w_0) < 0.5 \\ 1 & if & g(wx + w_0) > 0.5 \end{Bmatrix}$$

Where $g(\cdot)$ Is the Sigmoid activation function

The third and final classification model implemented is a Random Forest (RF) classifier. This method of classification is more complex than SVM and LRM due to the number of operating agents or trees established for classification upon startup. The output of a Random Forest model is selected by a voting algorithm selecting the most common result at the end of the random tree algorithm. This method is much more robust to overtraining compared to the logistic regression method as it features “drop-out” components to ignore certain features in each established tree.

The flow chart diagram in Figure 5 highlights the methods taken in flowchart steps for evaluation.

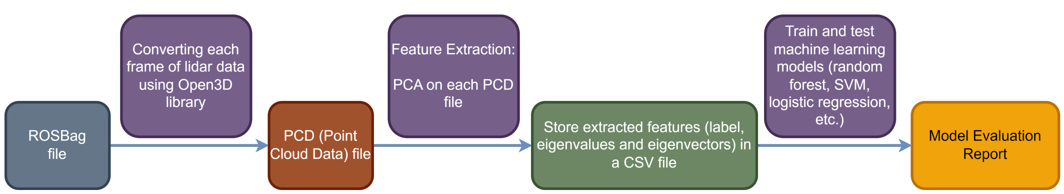

Figure 5: Flowchart outlining the methods implemented in this paper.

3 Results

Preliminary results highlight the relative performance of each of the three implemented classification algorithms accuracy relative to the initial preprocessing methods undertaken.

The total accuracy for each preprocessing method is compared for each of the classification algorithms. Table 3 outlines the results for each of the pattern recognition methods with respect to each specific method of pre-processing.

Table 3: Pattern Recognition Method Accuracy with Respect to Preprocessing Methods.

Pre-processing Method	Pattern Recognition Method Accuracy		
	Random Forest	SVM	Logistic Regression
RAW data	0.804	0.799	0.769
Window Cropping	0.804	0.697	0.738
Depth Cropping	0.8072	0.794	0.772
Voxel Grid Downsampling	0.811	0.832	0.831

For each of the classifiers, Voxel Grid Downsampling performed the best in overall accuracy relative to other preprocessing methods. Figure 6 showcases the relative accuracy of each of the classifiers with the voxel downsampling preprocessing applied. The SVM classifier performed best, with only a marginal (tenth of a percentage) increase over logistic regression.

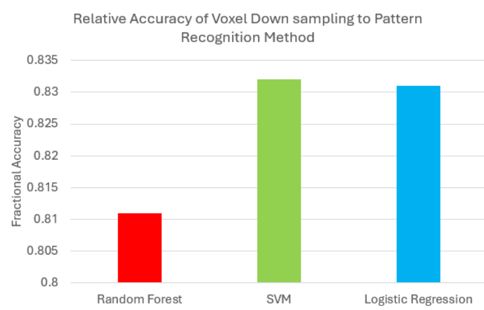

Figure 6: Relative accuracy of pattern recognition methods with downsample preprocessing

The relative accuracy of the pattern recognition implemented in predicting false positives (affecting overall accuracy) is not crucial. Instead, comparing recall can be used to assess the relative accuracy in the algorithm as it pertains to false negatives. Ideally a higher recall indicates that fewer true positives are being missed (conditions for ascending stair detection), which is critical in determining stairs are in frame. Recall is calculated as:

True Positive/(True Positives + False Negatives)

Voxel based downsampling had the best recall values for all preprocessing methods. Figure 7 showcases recall accuracy vs. Classifier algorithm. Logistic Regression has the highest recall by a much larger percentage compared to SVM.

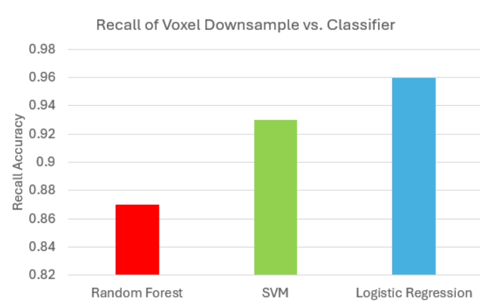

Figure 7: Recall accuracy of each pattern recognition method with downsampling

4 DISCUSSION

Preliminary results show the possibility that early model free pattern recognition based detection of ascending stairs is possible. The classification algorithms implemented here are intended to be fast operating evaluation methods that may take in frame by frame data and evaluate for ascending stairs within the desired detection range of 1 metre, and 90 degree arc from the centre of the bottom stair.

Preprocessing methods were investigated to determine the effect that each had on the potential of influencing processing time, and algorithm quality. Voxel downsampling methods influenced results the most, most likely due to the removal of extraneous, noisy, and irrelevant data points without sacrificing characteristic stair information. With downsampling, it is more likely to guarantee that scene information with evenly spaced stair points will influence PCA the most.

Logistic regression and SVM pattern recognition models implemented have similar accuracy for testing data (83.1 %,83.2% respectively). This relative accuracy highlights that both methods are comparable in both classification, and because both are single operation classifiers run time is essentially equivalent. Logistic regression, however, proves to be much more effective in recall accuracy, likely because of the non-linear decision boundary that may be generated, as compared to the linear SVM decision boundary kernel selected.

Recall may be used as an indicator of a better performing classifier when relative difference in overall accuracy is minor. Recall is a good indicator of how likely the system is to miss identifying stairs. The logistic regression classifier has an accuracy of 96% when it comes to successfully identifying stairs in a scene if they are within the desired identification range. SVM recall is 93%, so while not necessarily much worse, for the purpose of identifying stairs in the scene the logistic regression model performs better.

4.1 FUTURE STEPS

Preprocessing methods directly influence the quality of the LiDAR data and scene information, and thus have a relatively large impact on classifier accuracy. Investigating alternative methods for preprocessing and altering LiDAR data would focus mostly on methods similar to voxel based downsampling, including time-based downsampling methods to smooth points overtime [ref]. If processed over a small enough interval relative to user motion, this method should be able to construct less variable scene information by averaging points within voxels overtime, instead of just within a scene.

Feature detection engineering may allow for identification of patterns within scene data that PCA does not contain. Adjustments to current features and investigating alternative features taken from the LiDAR data may improve accuracy. Model specific information such as planes or lines/vertices were avoided as features in this study, however may provide value in identifying stairs in the scene. Plane dimensions, average relative positions, headings, etc. may provide features that help remove edge cases that prevent higher accuracy currently without greatly increasing computation time. Alternatively, feature-free methods of classification such as Deep Neural Nets provide some value in classification to identify features without the need of feature engineering.

Alternative classification methods, such as kNNs or maximum likelihood estimates may provide estimates of similar quality and speed as the SVM and Logistic Regression node, however a more interesting opportunity is presented in applying additional classification or fuzzy logic controllers onto classifier outputs over a recent time basis to create artificial memory. By using a voting system similar to random forest methods over a period of time, we can treat a series of prior estimates seeing “stairs” 80% of the sliding time window as confirmation that stairs are 100% in-scene. The threshold for this voting may be treated similar to a sigmoidal activation function, and may even be tuned as an additional simple layer. Over a period of 1 second, this method completely eliminates error in the presented case, however most collections are performed statically or at very low translational velocities.

Lastly, there are plans for additional collections to add a set of scenes including descending and alternative stair designs. Classification methods have proven to work on stairs other than uniform ascending stairs in creating traversal maps and so identifying curving ascending stairs and descending stairs is the next application.

Author Contributions

Christian Mele and Shovon Saha contributed equal time in the completion of this paper. Writing, presentation design and methods design (algorithms, feature detection) were performed by Christian. Code writing and algorithm implementation (algorithms, preprocessing, feature detection) were lead by Shovon. Collection design was lead by Shovon but carried out by both authors.

Acknowledgments

Thank you to Dr. Ali Ayub for inspiring and informing the content of this paper. It would not have been possible without the content of his lectures to inform how to approach scene detection and binary classification.

References

Hannes Harms, Eike Rehder, Tobias Schwarze, and Martin Lauer. Detection of ascending stairs using stereo vision. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2496–2502. IEEE, 2015.

Weiguang Huo, Samer Mohammed, Juan C Moreno, and Yacine Amirat. Lower limb wearable robots for assistance and rehabilitation: A state of the art. *IEEE systems Journal*, 10(3):1068–1081, 2014.

Jinrae Kim, Sunggoo Jung, Sung-Kyun Kim, Youdan Kim, and Ali-akbar Agha-mohammadi. Staircase localization for autonomous exploration in urban environments. *arXiv preprint arXiv:2403.17330*, 2024.

Andrew Garrett Kurbis, Brokoslaw Laschowski, and Alex Mihailidis. Stair recognition for robotic exoskeleton control using computer vision and deep learning. In *2022 International Conference on Rehabilitation Robotics (ICORR)*, pp. 1–6. IEEE, 2022.

Prasanna Sriganesh, Namya Bagree, Bhaskar Vundurthy, and Matthew Travers. Fast staircase detection and estimation using 3d point clouds with multi-detection merging for heterogeneous robots. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9253–9259. IEEE, 2023.

Jing Wang and Kuangen Zhang. Unsupervised domain adaptation learning algorithm for rgb-d staircase recognition. *arXiv preprint arXiv:1903.01212*, 2019.