Environmental Context Prediction for Lower Limb Prostheses With Uncertainty Quantification

Boxuan Zhong, Rafael Luiz da Silva, *Student Member, IEEE*, Minhan Li, He Huang, *Senior Member, IEEE*, and Edgar Lobaton, *Member, IEEE*

***Abstract*— Reliable environmental context prediction is critical for wearable robots (e.g., prostheses and exoskeletons) to assist terrain-adaptive locomotion. This article proposed a novel vision-based context prediction framework for lower limb prostheses to simultaneously predict human's environmental context for multiple forecast windows. By leveraging the Bayesian neural networks (BNNs), our framework can quantify the uncertainty caused by different factors (e.g., observation noise, and insufficient or biased training) and produce a calibrated predicted probability for online decision-making. We compared two wearable camera locations (a pair of glasses and a lower limb device), independently and conjointly. We utilized the calibrated predicted probability for online decision-making and fusion. We demonstrated how to interpret deep neural networks with uncertainty measures and how to improve the algorithms based on the uncertainty analysis. The inference time of our framework on a portable embedded system was less than 80 ms/frame. The results in this study may lead to novel context recognition strategies in reliable decision-making, efficient sensor fusion, and improved intelligent system design in various applications.**

***Note to Practitioners*—This article was motivated by two practical problems in computer vision for wearable robots: First, the performance of deep neural networks is challenged by real-life disturbances. However, reliable confidence estimation is usually unavailable and the factors causing failures are hard to identify. Second, evaluating wearable robots by intuitive trial and error is expensive due to the need for human experiments. Our framework produces a calibrated predicted probability as well as three uncertainty measures. The calibrated probability makes it easy to customize prediction decision criteria by considering how much the corresponding application can tolerate error. This study demonstrated a practical procedure to interpret and improve the performance of deep neural networks with uncertainty quantification. We anticipate that our methodology could be extended to other applications as a general scientific and efficient procedure of evaluating and improving intelligent systems.**

***Index Terms*— Bayesian neural network (BNN), environmental context prediction, prosthesis, uncertainty quantification.**

I. Introduction

Reliable environmental context prediction is critical for wearable robots (e.g., prostheses and exoskeletons) to better assist terrain-adaptive locomotion. For example, environmental context information can be used to predict human locomotion mode transition in advance for smooth lower limb prostheses control. Traditional approaches detected the locomotion mode transition using mechanical measurements [2], [3] (e.g., forces and motions), electromyography (EMG) signals [4], or inertial measurement units (IMUs) [5]—which are usually delayed, user-dependent, and sensor location sensitive.

Progress has been made to capture environmental context information for wearable robots. A laser distance meter and IMU-based system were developed in our previous studies [6], [7] to identify the terrain in front of the human, which was then fused with the human's neuromuscular signals to predict the user's locomotor task transitions. The single-point information from the laser distance meter is, however, inadequate to capture enough environmental features. Meanwhile, computer vision has attracted considerable attention because it is informative, noninterrupting, and user-independent. Depth images were used to detect user movement intent or recognize terrains (e.g., stairs and ramp) by edge detection [8]–[10] or 3-D point cloud classification [11]. Attempts have also been made to recognize terrains by learning from informative RGB images [12], [13]. Previous works showed promising results for scenarios with controlled variations in the environment and human behaviors. Studying variations in the real-life environment and human behaviors is necessary for wearable robotics applications. Moreover, a reliable confidence estimation is desired to quantify the uncertainty in the predictions of the vision algorithms, which was important for wearable robots to decide the competence of the algorithms and prevent potential risks to the human users.

Deep learning has shown promising potential in various vision applications, such as semantic segmentation [14] and site recognition [15]. In practice, the performance of the neural networks is challenged by data ambiguity, data noise,

Manuscript received March 20, 2020; accepted April 24, 2020. This article was recommended for publication by Associate Editor H. Ren and Editor M. Zhang upon evaluation of the reviewers' comments. This work was supported by the National Science Foundation under Award 1552828, Award 1563454, and Award 1926998. *(Corresponding authors: He Huang; Edgar Lobaton.)*

Boxuan Zhong, Rafael Luiz da Silva, and Edgar Lobaton are with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA (e-mail: bzhong2@ncsu.edu; rdasilv2@ncsu.edu; edgar.lobaton@ncsu.edu).

Minhan Li and He Huang are with the Joint Department of Biomedical Engineering, North Carolina State University, Raleigh, NC 27695 USA, and also with The University of North Carolina at Chapel Hill, Chapel Hill, NC 27599 USA (e-mail: mli37@ncsu.edu; hhuang11@ncsu.edu).

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

Digital Object Identifier 10.1109/TASE.2020.2993399

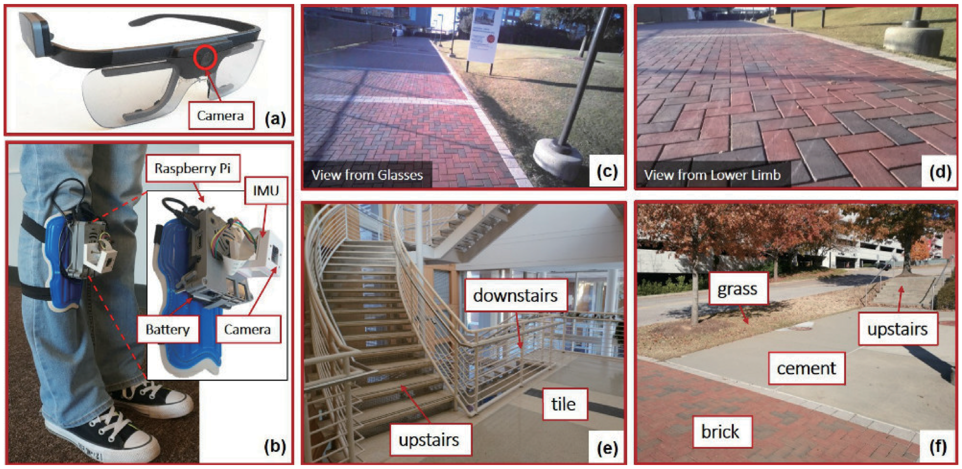


Fig. 1. Imaging devices and environmental context. (a) On-glasses camera configuration using a Tobii Pro Glasses 2 eye tracker [1]. (b) Lower limb data acquisition device with a camera and an IMU chip. (c) and (d) Example frames from the cameras for the two data acquisition configurations. (e) and (f) Example images of the data collection environment and terrains considered in our experiments.

and insufficient or biased training. For wearable robots, even occasional mistaken actions can lead to injuries. Accommodating actions (e.g., alarming the user or switching to a default "safe" mode) can be taken if the systems are aware of the low reliability of the current predictions. Thus, accurate uncertainty measures of the predictions are desired. There are two main types of uncertainty a model can have: epistemic and aleatoric uncertainty. Epistemic uncertainty captures the uncertainty of the model—because of a lack of understanding of the data generating mechanism, the model parameters are poorly determined, and the posterior over parameters are broad. Epistemic uncertainty is usually caused by insufficient or biased training and can be explained away with more training data. For example, if a terrain classifier is only trained with indoor scenes, large epistemic uncertainty is expected for outdoor scenes. Aleatoric uncertainty is usually caused by observation noises or insufficient sensing capability—for example, motion blur or overexposure for images. Aleatoric uncertainty is usually heteroscedastic (data-dependent) and cannot be reduced with more training data.

Prevailing deep learning models represent model parameters as deterministic values and do not capture epistemic model uncertainty. In addition, the observation noise is usually modeled as homoscedastic (not data-dependent) uncertainty and ignored as part of the weight decay. Recently, Gal and Ghahramani [16] proposed a dropout-based Bayesian approximation method to combine the advantages of the Bayesian theory and the state-of-the-art deep neural networks, leading to deep Bayesian neural networks (BNNs). This method was adapted to more complex neural network architectures, such as convolutional neural networks (CNNs) [17] and recurrent neural networks (RNNs) [18]. This method was then extended to capture both heteroscedastic aleatoric and epistemic uncertainty in a unified model [19]. The capability of capturing uncertainty provides BNNs with significant potentials in safety-sensitive fields, such as medical applications [20] and autonomous driving [21]. Although promising correlations with prediction errors were observed, the uncertainty measures of BNNs are not calibrated. Furthermore, the performance of BNNs for wearable robotics applications is unclear. Previous studies only applied BNNs to standard classification and regression tasks, such as depth estimation and semantic segmentation [19]. For wearable robots, user behavior is a significant source of uncertainty that does not exist in the previously studied applications. Moreover, it is an open question on how to verify and improve an intelligent system with the uncertainty measures from BNNs.

Another challenge of applying computer vision to wearable robots is the requirement for portability, which tightly constrains the sizes and locations of the sensors and processors. Various sensor locations have been used in present designs of wearable robots to capture environmental information, including chest [12], leg [9], [11], [13], and waist (side [6] or front [10]). In the previous studies, however, the choices of sensor locations were usually arbitrary, which warrants for a more formal study of the benefits of different sensor locations.

In this study, we proposed a novel environmental context prediction framework for lower limb prostheses, which can: 1) predict (simultaneously for multiple forecast windows) the type of terrain, which the user will be stepping on based on the current video streams from the wearable cameras; 2) quantify predictive uncertainties from different perspectives for interpretation; and 3) produce a calibrated predicted probability for online decision-making and fusion. In this study, we considered six types of terrains (tile, cement, brick, grass, upstairs, and downstairs as shown in Fig. 1) and four forecast windows (current, 1s, 2s, and 4s). Our framework can be extended to recognize more complex contexts, such as driving, sitting on the bus, and walking on a sidewalk with pedestrians and vehicles around. Other potential wearable robotic applications include: 1) notifying users in real time with context-aware suggestions for safety or rehabilitation and 2) autonomous patient condition analysis (e.g., analyzing gait behaviors on different terrains).

In addition, we evaluated and compared different wearable camera locations (see Table I) and fusion strategies

TABLE I

WEARABLE CAMERA LOCATION COMPARISON

Camera Locations	Advantages	Disadvantages
Glasses (Fig. 1 (a)).	(1) Captured videos are informative enough to support multiple environmental context prediction tasks (e.g. terrain prediction, egocentric action recognition [22], site recognition [15]); (2) The design is compatible with commercial smart glasses (e.g. Google glasses) and can be reused for multiple purposes (e.g. prosthesis control, rehabilitation instruction or entertainment).	(1) Not self-contained and constant communication with the prosthesis is required; (2) more irrelevant data exist because the head orientation is less controlled, and this perspective is more sensitive to view contamination in crowded scenes.
Lowerlimb (Fig. 1 (b)).	(1) The design can be integrated in prostheses, decreasing the cost of powering and communication; (2) Being closer to the terrains, the camera is able to capture more terrain features (e.g. texture, reflection) and less disturbances (e.g. surrounding pedestrian and vehicles);	(1) Camera is moving together with the lower limb, which introduces motion artifacts; (2) The prosthesis cannot be covered by pants—the camera needs to be unblocked.

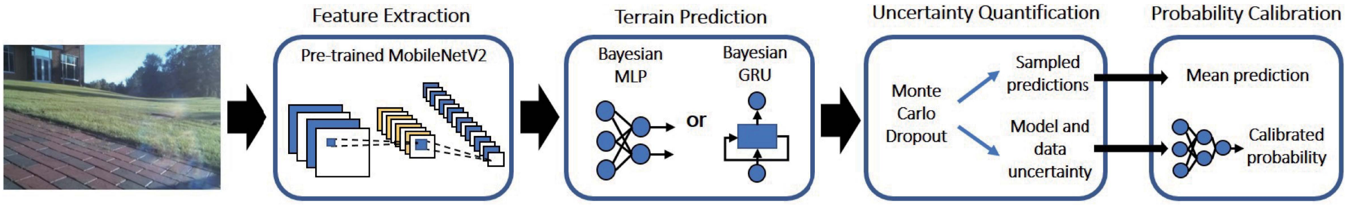


Fig. 2. Pipeline of the environmental context prediction framework. The framework reads the videos from the data acquisition devices (e.g., glasses and lower limb camera) and simultaneously outputs the terrain prediction and the calibrated predicted probability for different forecast windows.

from practical perspectives, such as prediction performance, generalization, and developing costs. We designed an evaluation methodology to verify and interpret the proposed framework considering the uncertainty measures, calibrated predicted probability, and the performances for different forecast windows under realistic disturbances. By leveraging the analysis, we developed a frame selection strategy that significantly reduced computations with similar prediction accuracy. We demonstrated a complete procedure of interpreting deep learning models with uncertainty measures, utilizing calibrated predicted probability for online decision-making and fusion, and closing the loop by improving the algorithms based on the uncertainty analysis. The results in this study may lead to novel context recognition strategies for reliable decision-making, efficient sensor fusion, and improved intelligent system design.

II. METHOD

A. Framework Overview

Fig. 2 shows the main components of our framework: feature extraction, terrain prediction, uncertainty quantification, and probability calibration.

First, we used MobileNetV2 [23] pretrained with the ILSVRC data set [24] to extract features from images. We adopted the implementation from Keras and extracted the features after the global average pooling layer but before the last Softmax layer. MobileNetV2 is popular for low computational cost and high prediction accuracy. We resized the images to 224 × 224 × 3 before feeding them into the MobileNetV2.

Second, trained a BNN to predict current and future terrains based on the features extracted by the MobileNetV2. During inference, we performed the Monte Carlo dropout sampling for 40 times to obtain the terrain prediction and the three uncertainty measures: aleatoric uncertainty, entropy uncertainty, and mutual information uncertainty. We studied two BNN architectures: Bayesian multilayer perceptron (BMLP) and Bayesian gated recurrent unit (BGRU). BMLP had three fully connected layers, and BGRU had one variational gated recurrent unit layer between two fully connected layers. For both networks, the first two layers had 512 units and used ReLU activation function. For each forecast window, the last layer had two components: 1) a fully connected layer with six units and a Softmax activation function to classify the six types of terrains and 2) a fully connected layer with one unit and a SoftPlus activation function to predict the aleatoric uncertainty parameter σ. The last layers for different forecast windows shared the first two layers. We applied dropout to each layer with a dropout probability of 0.1. We also applied l_2 regularization to the bias and the kernel parameters with the weight set to 10^{-5}.

Finally, we trained a probability calibration network projecting the three uncertainty measures to one calibrated predicted probability. This network had three fully connected layers with 32, 64, and 1 units, respectively. The tanh activation function was used for the first two layers while Sigmoid for the last layer.

B. Bayesian Deep Neural Network

BNNs extend standard neural networks by modeling the distributions over the weight parameters. Denote the input of a BNN as x, the collection of the parameters as W, and the output as $f^W(x)$. The prior of W is usually assumed as a standard Gaussian distribution $\mathcal{N}(0, I)$ with a transformation (if needed). Given a training data set with $X = \{x_1, \ldots, x_N\}$ as the input (the images/video feed in our case) and $Y = \{y_1, \ldots, y_N\}$ as the targets (the terrain which the subject will step on) where N is the number of training observations, the Bayesian inference is used to find the posterior distribution $\mathbb{P}(W|X, Y)$. Then, the prediction for a new input sample x^* is

$$\mathbb{P}\big(y^*|x^*, X, Y\big) = \int \mathbb{P}\big(y^*|x^*, W\big)\,\mathbb{P}(W|X, Y)\,dW. \tag{1}$$

However, for deep neural networks, $\mathbb{P}(W|X, Y)$ is analytically intractable. One direction of solutions is to approximate $\mathbb{P}(W|X, Y)$ with a simple distribution $q^*_\Theta(W)$. Θ is the collection of parameters of $q^*_\Theta(W)$ and can be estimated by minimizing the Kullback–Leibler divergence to $\mathbb{P}(W|X, Y)$. We utilized the Monte Carlo dropout sampling [16] to approximate the prediction and measure the epistemic uncertainty. Denote by W_i as the parameters (a matrix of size $K_i \times K_{i-1}$) of the ith layer of a BNN, Θ_i as the variational parameters (a matrix of the same size as W_i), and p_i as the dropout probability for the ith layer. Then, $q^*_\Theta(W)$ can be defined as

$$\begin{aligned} W_i &= \Theta_i \cdot \mathrm{diag}\Big(\big[z_{i,j}\big]_{j=1}^{K_{i-1}}\Big) \\ z_{i,j} &\sim \mathrm{Bernoulli}(p_i). \end{aligned}$$

With the abovementioned formulation, standard dropout can be used as a Bayesian approximation.

In addition, the aleatoric uncertainty σ can be learned from the data as an output of the model [19]. By Monte Carlo dropout sampling $\hat{W} \sim q_\Theta(W)$, the sampled predictions and the aleatoric uncertainty measure can be obtained as $[\hat{y}, \hat{\sigma}^2] = f^{\hat{W}}(x)$. The loss function $\mathcal{L}(\Theta)$ (associated with the likelihood) to train BNN for regression is given by

$$\frac{1}{N}\sum_{n=1}^{N}\left[\frac{1}{2}\hat{\sigma}_n^{-2}||y_n - \hat{y}_n||^2 + \frac{1}{2}\log\big(\hat{\sigma}_n^2\big)\right]. \tag{2}$$

This loss has two terms: a mean square error tempered by σ and a regularization term of σ. The first term discourages the model from predicting very small σ to samples with large error, and the second term discourages predicting large σ to all the samples. During training, the neural network is trained to adapt the weighting of the prediction errors—assigning large σ to the most challenging samples that are usually contaminated by noise. No ground-truth value is needed to train for σ because it is trained implicitly through the loss attenuation.

Suppose we performed the Monte Carlo dropout sampling for T iterations for each sample, the prediction can be approximated via

$$\mathbb{E}(y) \approx \frac{1}{T}\sum_{t=1}^{T}\hat{y}_t \tag{3}$$

and the uncertainty is given by

$$U \approx \frac{1}{T}\sum_{t=1}^{T}\hat{y}_t^2 - \left(\frac{1}{T}\sum_{t=1}^{T}\hat{y}_t\right)^2 + \frac{1}{T}\sum_{t=1}^{T}\hat{\sigma}_t^2 \tag{4}$$

where the first two terms capture the epistemic model uncertainty and the second term captures the aleatoric uncertainty.

We extended the regression aleatoric uncertainty to our classification task by modeling the regression uncertainty of the logit vector—the output of the last layer before the Softmax activation function. We placed a Gaussian distribution over the logit vector as $\hat{z} \sim \mathcal{N}(y, \sigma^2)$, where $[y, \sigma^2] = f^W(x)$, f^W is the neural network, and x is the input data. The expected log likelihood for each training sample is then

$$\mathcal{L} = \log\Big\{\mathbb{E}_{\mathcal{N}(\hat{z};y,\sigma^2)}[\mathrm{Softmax}(\hat{z}_c)]\Big\} \tag{5}$$

where c is the ground-truth label of x.

Since (5) is analytically intractable, we approximated it by Monte Carlo integration. Denote $\hat{z}_k = f^{\hat{W}} + \sigma \cdot \epsilon_k$, where ϵ_k follows a standard Gaussian distribution. The loss function becomes

$$\mathcal{L} = \log\frac{1}{K}\sum_{k=1}^{K}\exp\left[\hat{z}_{k,c} - \log\sum_{c'}\exp\big(\hat{z}_{k,c'}\big)\right] \tag{6}$$

where K is the number of Monte Carlo sampling iterations and c' is the class index of the logit vector $\hat{z}$.

Given T the number of Monte Carlo dropout sampling iterations, the output Softmax vector p^* for inference can be approximated as

$$p^* \approx \frac{1}{T}\sum_{t=1}^{T}\mathrm{Softmax}\Big(f^{\hat{W}_t}(x^*)\Big). \tag{7}$$

For classification, we adopted two alternative uncertainty metrics: predictive entropy and mutual information. Predictive entropy [25] measures the uncertainty of deciding the class of a sample—large if the estimated distribution is "broad" over different classes and small if "sharp" on one of the classes. With the Monte Carlo dropout sampling, it was approximated as

$$\begin{aligned} H\big[y^*|x^*, X, Y\big] = &-\sum_{c'}\left(\frac{1}{T}\sum_{t=1}^{T}\mathbb{P}\big(y^* = c'|x, \hat{W}_t\big)\right) \\ &\cdot\log\left(\frac{1}{T}\sum_{t=1}^{T}\mathbb{P}\big(y^* = c'|x, \hat{W}_t\big)\right). \end{aligned} \tag{8}$$

Mutual information between the posterior over W and the prediction y^* quantifies the uncertainty in the BNN's output [26]. This measure is larger when the stochastic predictions are less constant. It was calculated via

$$\begin{aligned} I\big[y^*, W|x^*, X, Y\big] = &\; H\big[y^*|x^*, X, Y\big] \\ &+\frac{1}{T}\sum_{c',t}\big[\mathbb{P}\big(y^* = c'|x, \hat{W}_t\big) \\ &\cdot\log\big(\mathbb{P}\big(y^* = c'|x, \hat{W}_t\big)\big)\big]. \end{aligned} \tag{9}$$

Predictive entropy captures both epistemic and aleatoric uncertainty, while the mutual information captures the epistemic model uncertainty [27].

C. Probability Calibration

Intuitively, an event of probability 0.75 should occur 75% of the time. Given a multiclass classifier with a prediction and corresponding predicted probability $[\hat{y}, \hat{p}] = \mathcal{H}(x)$ for an input x, a sufficient condition for calibration is

$$\mathbb{P}(\hat{y} = y|\hat{p} = p) = p, \quad \forall p \in [0, 1]. \tag{10}$$

Unfortunately, the three uncertainty measures in our framework were not calibrated, which is a common issue for

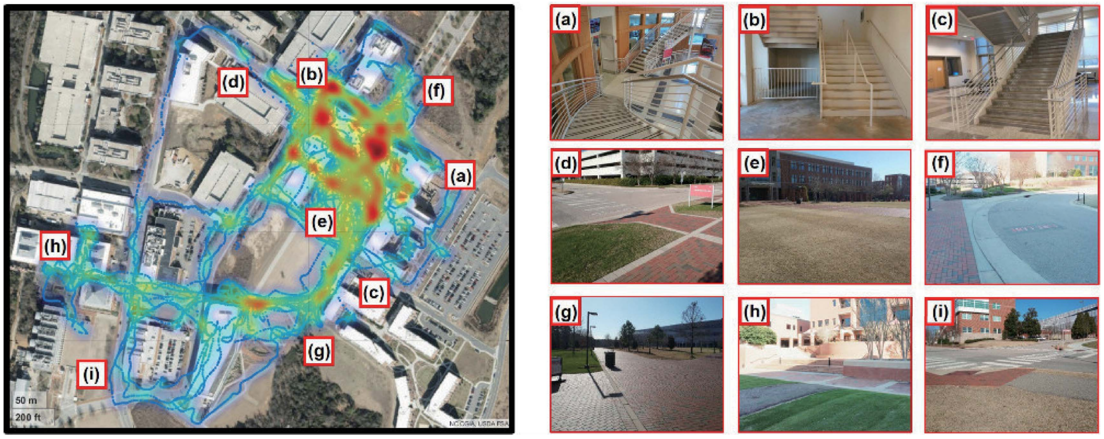


Fig. 3. Geographic density map of our data set with example images of the indoor and outdoor sites. The grass and brick can be found in (d)–(i) images; tile in (a) and (c) images; stairs in (a)–(c) and (h) images; and cement in (b), (d), (f), and (i) images.

BNNs [28]. A calibration function $\Psi : \mathbb{R}^3 \to [0, 1]$ is desired such that $\Psi \circ M$ produces calibrated probability, where M represents the three uncertainty measures from the BNN. In our framework, we trained a neural network to approximate Ψ.

D. Vision Fusion

We used two vision fusion strategies to combine the information from the glasses and the lower limb camera.

1) *Feature fusion:* After extracting image features with the pretrained network (see Fig. 2), the feature vectors for glasses and lower limb were concatenated before fed into the terrain prediction network.
2) *Decision fusion:* The terrain prediction and probability calibration network (see Fig. 2) were first trained for glasses and lower limb separately. During inference, the predictions with the higher predicted calibrated probability were taken as the fused predictions.

Compared with feature-level fusion, decision-level fusion is more favorable in practice because it does not require collecting expensive synchronized training data with multiple devices. It is possible to aggregate the data sets collected by different users from multiple locations with only one of the devices, creating the opportunity to employ crowd-sourcing strategies.

III. Experimental Method

A. Experiment Setup and Protocol

All participants provided informed, written consent to participate in our research approved by the Institution Review Board (IRB) of The University of North Carolina at Chapel Hill and North Carolina State University. Seven able-bodied participants (five males and two females) and one transtibial amputee (male) participated in this study. Their ages were between 20 and 60 years old. During the experiment, participants wore the Tobii Pro Glasses 2 eye tracker [1] [see Fig. 1(a)] and a lower limb device [see Fig. 1(b)]. The eye-tracker recorded videos at 25 frames/s (FPS) and a resolution of 1240 × 1080. Videos were recorded by the camera at the center of the glasses and stored locally in the SD card. The lower limb device was attached to the shins of the able-bodied participants. For the amputee subject, the device was attached on top of the pants around the socket of passive lower limb prosthesis. The device was made of a Raspberry Pi 3 Model B, a PiCamera, and an Adafruit BNO055 IMU chip. The PiCamera recorded video at 25 FPS with a resolution of 1240 × 1080 and the IMU chip recorded the accelerometer and gyroscope data at a rate of around 100 Hz. Data were recorded from the IMU chip and the camera and stored to an SD card in the Raspberry Pi. For accurate and fast labeling, we used another stand-alone camera to record the terrain context of the participants. We did not use the videos from the glasses or the lower limb device for labeling because the videos were sometimes blurry. At the beginning and the end of each data collection session, we let the three cameras record a running timer simultaneously. We synchronized the three videos by finding the frames of the timer with the same displayed number. The participants were instructed to walk according to their own will on NC State University Centennial campus. During the experiments, we also recorded the GPS coordinates of the participants. Fig. 3 shows the geographic density map of our data set. Fig. 3(a)–(c) show the examples of indoor sites, and Fig. 3(d)–(i) show the examples of outdoor sites. The data set totals around 11 h of recording. The source code and data are available online.[1] After sampling to 10 FPS, the data set had around 327 000 RGB images—69 000 for tile, 55 000 for grass, 110 000 for brick, 59 000 for cement, 19 000 for upstairs, and 15 000 for downstairs.

B. Training and Testing Procedure

The entire data set was divided into training, validation, and testing data sets. The results in Section IV-D are based on the data of the amputee, while the results in other sections are based on the data of the able-bodied participants.

We performed the leave-one-participant-out cross-validation for three able-bodied participants because they had a similar amount of data—each participant had four sessions of data

[1] https://research.ece.ncsu.edu/aros/paper-tase2020-lowerlimb

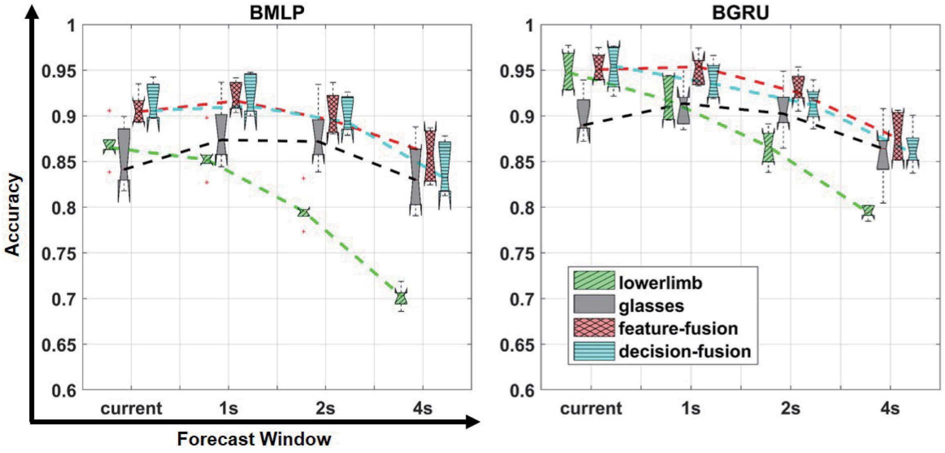


Fig. 4. Results of the terrain prediction: lower limb and glasses indicate the results of utilizing only one of the two data acquisition devices; feature fusion and decision fusion indicate the results of fusing both devices.

and each session had 20∼25 min of video. For each iteration of evaluation, we had around 244 000 RGB images in the training data set, around 40 000 in the validation data set and 43 000 in the testing data set. In addition, around half of the testing videos were collected during busy hours with more pedestrians in the cameras' field of view. We verified the difference between the two groups of videos by detecting people in the videos from the glasses camera with an object detection algorithm [29]. People can be detected in 52.9% of the frames for videos during busy hours while 5.2% for the other videos. We analyzed the effect of this disturbance later in Section IV-B.

We collected around 35 min of video (21 000 images after downsampling to 10 FPS) for the amputee and did not collect videos during busy hours for safety. We used the data of the able-bodied participants for the training and validation data set and the data of the amputee for the testing data set.

We trained the terrain prediction network and the calibration network (see Fig. 2) in three steps. First, the terrain prediction network was trained with the training data set. Second, we generated the Monte Carlo dropout predictions and uncertainty measures for the validation data set with the trained terrain prediction network. Third, if one sample was mistakenly/correctly predicted in step two, we labeled it as 0/1. These labeled samples were used to train the calibration network whose input was the uncertainty measures and ground-truth output were the 0/1 labels. Binary cross-entropy loss was used to train the calibration network. Finally, we conducted the evaluation and analysis with the testing data sets. A video demonstrating our experiments is available online.[2]

C. Probability Calibration Diagnosis

Reliability diagram [30] is a common visual tool to evaluate model calibration. The diagram plots the observed frequency of an event against the estimated probability of this event. In our situation, the event was defined as successful terrain predictions. Given a data set $\{(x_n, y_n)\}_{n=1}^{N}$ of size N, $[\hat{y}_n, \hat{p}_n] = \mathcal{H}(x_n)$ were the prediction and predicted probability. We grouped the predictions into K interval bins of size $(1/K)$. Given $I_k = (((k-1)/K), (k/K)]$ as the kth interval, the number of samples belong to I_k was $B_k = \sum_{n=1}^{N} \mathbb{1}\{\hat{p}_n \in I_k\}$. Then, the observed classification accuracy for I_k was

$$\mathrm{acc}_k = \frac{1}{B_k} \sum_{n=1}^{N} \mathbb{1}\{\hat{y}_n = y_n, \hat{p}_n \in I_k\}$$

which was a consistent and unbiased estimator of $\mathbb{P}(\hat{Y} = Y | \hat{P} \in I_k)$. The mean predicted probability for I_k was defined as

$$\mathrm{prob}_k = \frac{1}{B_k} \sum_{\hat{p}_n \in I_k} \hat{p}_n.$$

For a perfect calibrated model, $\mathrm{prob}_k = \mathrm{acc}_k$ for all $k \in \{1, 2, \ldots, K\}$.

Expected calibration error (ECE) and maximum calibration error (MCE) [31] are two statistical metrics of miscalibration calculated as

$$\mathrm{ECE} = \frac{1}{N} \sum_{k=1}^{K} B_k |\mathrm{acc}_k - \mathrm{prob}_k| \tag{11}$$

and

$$\mathrm{MCE} = \max_{k=1}^{K} |\mathrm{acc}_k - \mathrm{prob}_k|. \tag{12}$$

For a perfect calibrated probability, ECE and MCE equal to 0, and larger values indicate worse calibration.

IV. Result and Discussion

A. Performance Comparison

The lower limb camera had a better performance when predicting closer terrains, while the glasses were better at predicting further-away terrains. Fusing both cameras achieved the best performance.

Fig. 4 presents the performances of different camera locations and prediction methods. The performance of the lower limb camera decreased while predicting further into the future.

[2] https://youtu.be/Cly0PJx9Gz4

TABLE II

AVERAGE TERRAIN PREDICTION ACCURACY AND STANDARD DEVIATION (IN BRACKETS). HIGHEST (DARK BLUE) AND SECOND HIGHEST (LIGHT BLUE) ACCURACY ARE HIGHLIGHTED

		Overall Accuracy (%)				Transition Period Accuracy (%)			
Device	**Method**	**current**	**1s**	**2s**	**4s**	**current**	**1s**	**2s**	**4s**
lowerlimb	BMLP	86.86(1.98)	85.57(2.11)	79.72(1.74)	70.19(1.02)	80.67(2.40)	77.43(2.96)	64.52(2.66)	43.32(2.10)
	MLP	86.78(2.04)	85.85(1.88)	79.52(1.43)	69.43(1.16)	80.14(1.95)	77.52(2.43)	63.93(2.29)	42.39(2.05)
	BGRU	**94.97(2.02)**	91.89(2.15)	86.55(1.81)	**79.46(0.62)**	**93.09(2.15)**	**85.66(1.98)**	73.25(1.15)	**57.49(3.77)**
	GRU	94.53(2.19)	**92.10(2.05)**	**87.50(2.16)**	78.40(1.28)	93.01(2.20)	85.36(2.22)	**73.72(3.24)**	56.72(2.63)
glasses	BMLP	85.25(2.97)	88.12(3.04)	87.84(3.10)	83.41(3.42)	73.61(4.97)	78.51(4.73)	77.72(4.15)	70.56(4.55)
	MLP	84.50(2.91)	87.51(3.06)	87.14(2.92)	82.48(3.94)	73.47(5.12)	78.29(4.44)	77.34(3.82)	69.59(4.08)
	BGRU	*89.92(2.24)*	*91.30(2.24)*	*90.52(2.58)*	*85.92(3.18)*	*81.94(3.88)*	*83.09(3.34)*	*80.45(3.41)*	*71.96(5.61)*
	GRU	**91.10(2.40)**	**91.96(2.62)**	**90.87(2.42)**	**85.99(3.73)**	**83.74(4.04)**	**83.73(3.82)**	**81.05(3.04)**	**72.47(5.41)**
lowerlimb + glasses	BMLP-feature-fusion	90.78(1.48)	92.04(1.45)	90.18(2.09)	85.67(2.48)	84.33(2.61)	84.48(2.19)	79.85(3.33)	69.79(4.41)
	MLP-feature-fusion	90.92(1.99)	92.51(1.89)	91.17(1.96)	85.18(2.95)	84.75(2.54)	84.92(2.65)	80.29(2.69)	70.15(4.30)
	BGRU-feature-fusion	**95.36(1.40)**	**95.15(1.45)**	*92.88(1.57)*	**87.73(2.18)**	**92.54(1.99)**	**89.43(2.30)**	*82.89(2.58)*	*72.17(4.86)*
	GRU-feature-fusion	94.57(1.55)	*94.98(1.43)*	**93.46(1.73)**	*87.32(2.20)*	92.26(1.97)	*88.78(2.01)*	**83.92(2.82)**	**73.10(4.24)**
	BMLP-decision-fusion	91.41(1.78)	92.00(1.95)	89.95(1.78)	84.05(2.53)	83.65(2.69)	84.19(2.63)	78.85(2.58)	65.65(3.74)
	BGRU-decision-fusion	*95.23(2.17)*	93.84(1.88)	91.24(1.78)	86.40(2.03)	*92.40(1.91)*	88.06(1.99)	80.71(1.94)	69.24(3.74)

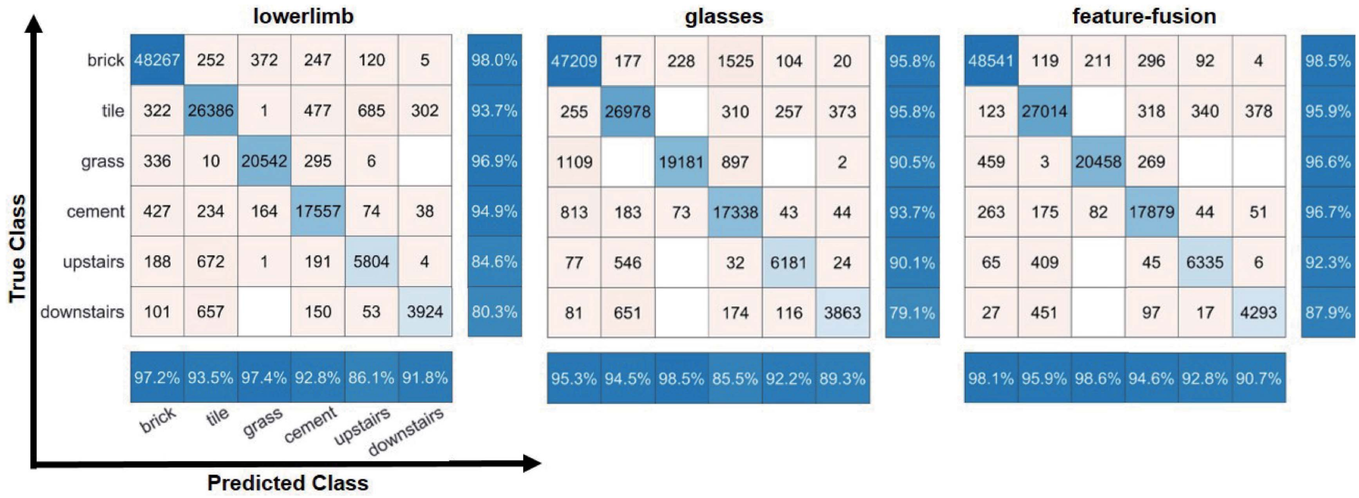


Fig. 5. Confusion matrix of terrain prediction with the two data acquisition devices individually and jointly (feature fusion). The results are for predicting the terrain 1 s in the future using the BGRU model. For each confusion matrix plot, the true positive rates and the positive predictive values are in the row and column summery, respectively.

The glasses camera achieved the best performance (up to 91.30% for BGRU) when predicting 1∼2 s into the future. It was better than predicting the current terrain (89.92% for BGRU) because the glasses camera did not always capture the immediate terrains. Feature fusion achieved the best performance (up to 95.36% for BGRU) since it jointly modeled the information from both cameras. In general, decision fusion gave slightly worse results than feature fusion—for BGRU and 1s forecast window, the accuracy was 93.84% for decision fusion while 95.15% for feature fusion. Decision fusion, however, has the benefit of not requiring synchronized data collection with multiple devices.

Table II shows the terrain prediction accuracy for different approaches. Given that the subject stepped on a new terrain at time t_s, we defined the transition period as the 5 s period before t_s, i.e., $[t_s - 5, t_s)$. The transition period accuracy was the terrain prediction accuracy during the transition periods. We reported the average and standard deviation of the accuracy. In general, BNNs showed similar prediction accuracy as the standard neural networks since the BNN in our framework inherited the merits of modern deep neural networks while enhancing them with reliable and interpretable uncertainty measures. Compared with lower limb, glasses showed a larger standard deviation in performance because human head movements usually have larger variations. Predicting during transition periods was more challenging (lower accuracy than the overall accuracy) because it required accurate estimation of the walking speed, direction, and distance to the terrain boundary.

Fig. 5 shows the confusion matrices of terrain prediction. The results are for predicting 1 s into the future using BGRU. For each plot, the true positive rates and the positive predictive values are in the row and column summary, respectively. Identifying upstairs and downstairs was more challenging (lower true positive rates and positive predictive values) compared with the other four terrains because: 1) our data set had less training samples for upstairs and downstairs and 2) it was difficult for the cameras to capture stairs. The lower limb camera's view was often limited to partial or one staircase, while the glasses could hardly capture the downstairs. Compared with using one camera, fusing both cameras increased the performance by up to 10% for upstairs and downstairs.

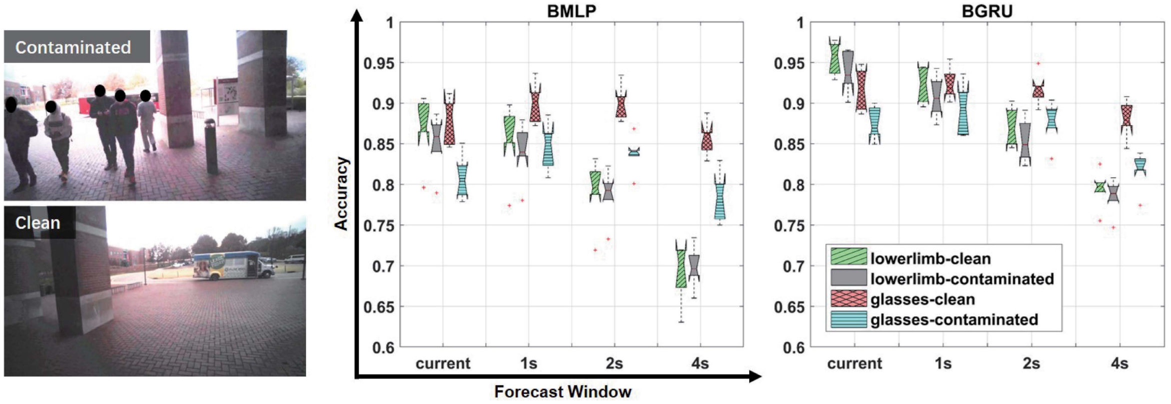


Fig. 6. Performance of the framework under one challenging scenario—camera view contamination caused by surrounding pedestrians. The images on the left are example frames captured by the glasses camera with the contaminated and clean views.

B. Performance With View Contamination

Surrounding pedestrians tended to compromise the performances of the glasses more than the lower limb camera. We observed a decrease in the prediction accuracy by around 1% for lower limb while around 5% for glasses.

Fig. 6 shows the performance of our framework under one realistic challenging scenario—camera view contamination caused by surrounding pedestrians. For BMLP and 1s forecast window, the prediction accuracy was 85% for glasses and 84% for lower limb with contamination, while the accuracy was 90% for glasses and 85% lower limb without contamination. The lower limb was more robust to this type of disturbance because lower limb focused on recording the terrain, while glasses captured images consisting of the information of the surrounding environment (such as buildings and sky), beyond just terrain information.

C. Uncertainty Analysis

We verified the model calibration (see Section IV-C1) and studied the correlations between uncertainty measures and prediction errors (see Section IV-C2). We also analyzed the uncertainty patterns for gait phases with the lower limb device (see Section IV-C3).

1) Model Calibration: Our predicted probability was well calibrated while the standard Softmax probability tended to be overconfident.

Fig. 7 shows the reliability diagrams of our calibrated probability from BNNs (left) and the Softmax probability from standard neural networks (right). The histograms at the bottom present the distributions of the predicted probabilities. Our predicted probability was able to better match the true accuracy, while the Softmax probability tended to be overestimated, which is a common issue for modern deep neural networks [32]. The Softmax probability of MLP-lower limb showed better calibration than the others. We assume that this is because less overfitting is present in the MLP-lower limb approach compared with the other methods since the videos from lower limb contained more variations than the videos from glasses, and compared with GRU, MLP had lower model complexity—no temporal relationship needs to be modeled. Overfitting has a negative impact on model calibration [32]. Table III reports the ECE and the MCE. Besides the trends that we observed in Fig. 7, the fusion methods tended to have better model calibration than methods using only one device and decision fusion showed better calibration than feature fusion.

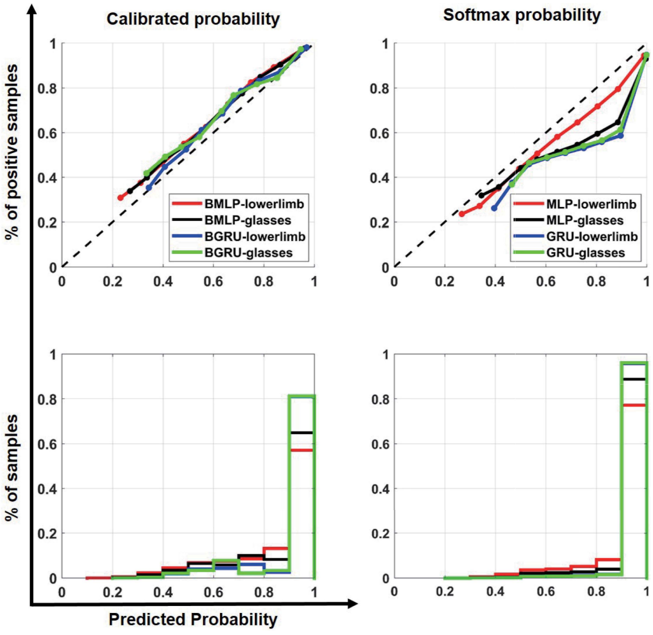


Fig. 7. Reliability diagrams (top) and predicted probability histograms (bottom) for our calibrated probability of BNNs (left) and the Softmax probability of standard neural networks (right). The results show an average of all forecast windows (i.e., current, 1s, 2s, and 4s).

We also studied the calibration performance for different combinations of input uncertainties. We modified the calibration function $\Psi : \mathbb{R}^k \rightarrow [0, 1]$ to map a subset of the three uncertainty measures to a calibrated probability, where $k = 1, 2, 3$. We have seven subsets of uncertainties to compare. We calculated the ECE/MCE for eight device–method combinations and all forecast windows: BMLP/BGRU-lowerlimb/glasses (four combinations) and BMLP/BGRU-feature/decision-fusion (four combinations). We then Z-score normalized the ECE/MCE values for each device–method

TABLE III

RESULTS OF ECE AND MCE. LOWEST (DARK BLUE) AND SECOND LOWEST (LIGHT BLUE) ERRORS ARE HIGHLIGHTED

Device	Method	ECE	MCE
lowerlimb	BMLP	0.0371	**0.0768**
	MLP	0.0535	0.0906
	BGRU	**0.0187**	0.0774
	GRU	0.0594	0.3089
glasses	BMLP	**0.0299**	**0.0747**
	MLP	0.0800	0.2387
	BGRU	0.0315	0.0860
	GRU	0.0596	0.2820
lowerlimb + glasses	BMLP-feature-fusion	0.0198	0.1311
	MLP-feature-fusion	0.0434	0.1627
	BGRU-feature-fusion	0.0138	0.1366
	GRU-feature-fusion	0.0405	0.2816
	BMLP-decision-fusion	0.0093	**0.0526**
	BGRU-decision-fusion	**0.0087**	0.1052

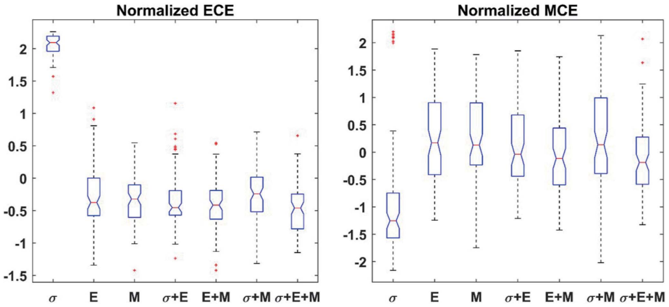


Fig. 8. Normalized ECE and MCE for different combinations of the input for the calibration network. "σ," "E," and "M" represent aleatoric, predictive entropy, and mutual information uncertainty, respectively.

combination and compared the summarized performances in Fig. 8. "σ," "E," and "M" represent aleatoric, predictive entropy, and mutual information uncertainty, respectively. Fusing all three uncertainty measures ("σ + E + M") achieved the best calibration performance. In general, entropy uncertainty is critical for a good calibration although entropy uncertainty itself is not enough and needs to be combined with aleatoric or mutual information uncertainty (see "E + M" and "E + σ"). In this study, we modeled the uncertainties based on a few assumptions, such as the Gaussian prior to the weights and the Gaussian aleatoric uncertainty on the logit vector. Our assumptions can approximate but not perfectly represent the actual data/noise/weight distributions. We assume that this is one of the reasons why fusing all three uncertainty measures produced the best calibration performance in this experiment.

The results in this section are based on an average of all forecast windows (e.g., current, 1s, 2s, and 4s). The results for each forecast window were similar.

2) Aleatoric and Epistemic Uncertainty: Compared with BMLP, BGRU showed lower aleatoric and epistemic uncertainty. In our experiments, epistemic uncertainty played a greater role than aleatoric uncertainty in detecting mistaken predictions, especially for BGRU.

Fig. 9 shows the distributions of different uncertainty measures for correct and mistaken terrain predictions. For both BMLP and BGRU, the calibrated probability could well distinguish the two groups with aleatoric uncertainty playing a less significant role than epistemic uncertainty.

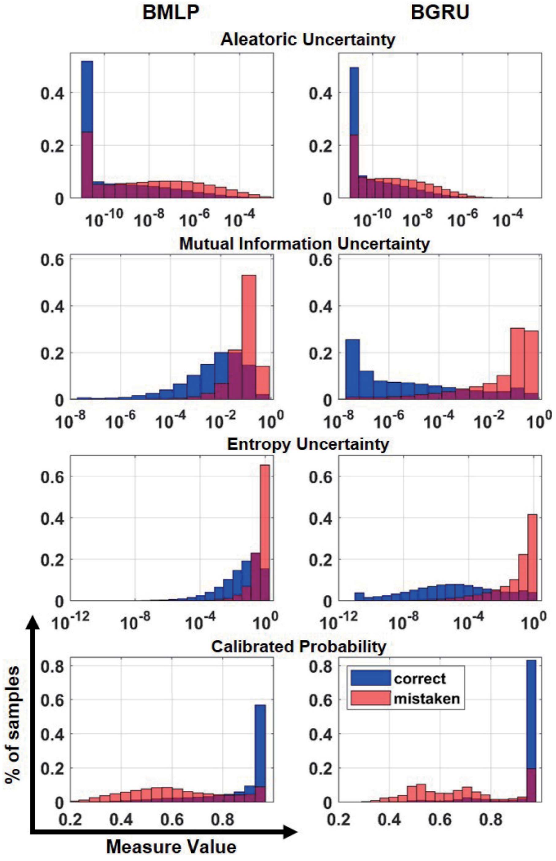
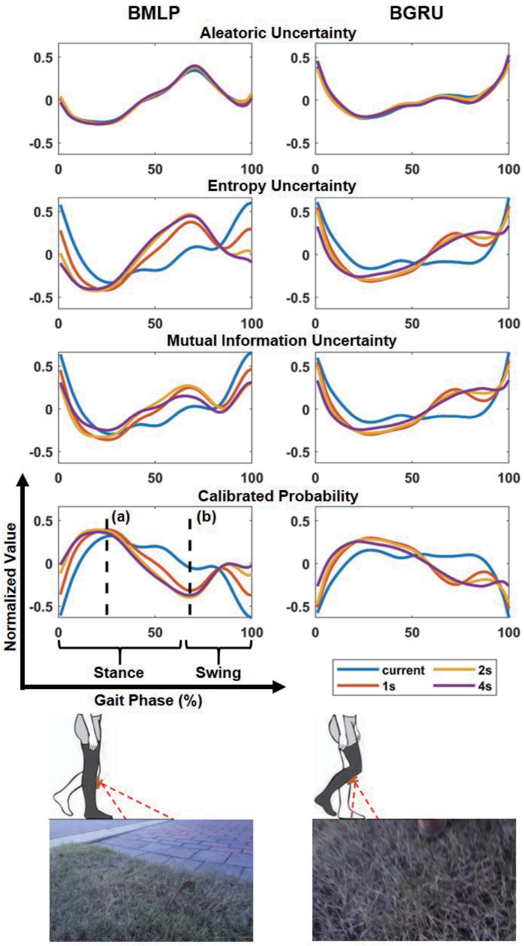


Fig. 9. Distributions of uncertainty measures and calibrated probability for correct and mistaken terrain predictions. Two neural network structures are involved: Bayesian MLP (left) and Bayesian GRU (right). The plots are based on the results for the lower limb camera.

With the information from multiple frames, BGRU decreased the aleatoric uncertainty from occasional observation noises. Compared with BMLP, BGRU produced lower entropy and mutual information uncertainty for correct predictions benefiting from additional temporal features (e.g., walking speed and direction). However, the entropy and mutual information uncertainty remained high for mistaken predictions that occurred under challenging scenarios (e.g., multiple terrains ahead or changing walking speeds). Due to the large variations in human behavior, the training data set was not large enough to explain away the epistemic uncertainty. Fig. 9 reports the averaged results of all forecast windows (i.e., current, 1s, 2s, and 4s) with lower limb . The results for the glasses camera and different forecast windows were similar.

3) Gait Analysis: The uncertainty measures showed periodic patterns for lower limb for gait cycles. In general, the framework showed the lowest uncertainty during the middle-stance gait phase.

The video and IMU signals from the lower limb device showed periodic patterns correlated with the lower limb walking motions. We analyzed the uncertainty measures using the normalized gait cycle by following these steps: 1) we used a peak detection algorithm (implemented by MATLAB [33]) to segment the gait cycles based on the gyroscope signals; 2) the uncertainty measures were Z-score normalized for each gait cycle; and 3) we normalized the gait cycles to the same length



(a) (b)

Fig. 10. Top: averaged uncertainty measures and calibrated probability with respect to the normalized gait cycle. The values are Z-score normalized for each gait cycle. Two neural network structures are compared: Bayesian MLP (left) and Bayesian GRU (right). (a) and (b) Two example frames taken by the lower limb camera and the dotted lines indicate the corresponding gait phases.

and then aligned and averaged the corresponding uncertainty measures. Fig. 10 shows the processed uncertainty measures aligned with the normalized gait cycle. We labeled the stance and swing phases with the IMU signals and the context videos. The images on the top are example frames during different phases indicated by the dotted lines. The results are the averaged data of all three able-bodied participants, and the results for each participant were similar.

For BMLP, the framework showed the highest calibrated probability (lowest uncertainty) in the middle of the stance phase [see Fig. 10(a)] because the camera was able to capture the current and future terrains with small irrelevant regions (e.g., sky or too far-away terrains). Moreover, less camera movement during this time led to less blurry images. During the early swing phase [see Fig. 10(b)], the captured images were blurry and tended to miss further-away terrains. As a result, the framework showed high uncertainty, especially for predicting further-away terrains. The aleatoric uncertainty showed similar trends for different forecast windows because it focused on measuring the occasional observation noises (e.g., motion blur and overexposure) in the input data.

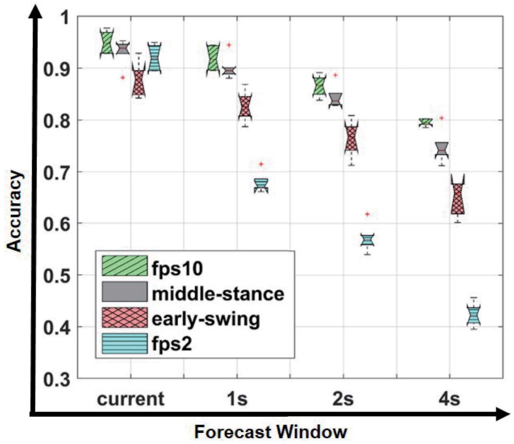


Fig. 11. Results of the terrain prediction with different frame selection strategies. The fps10 and fps2 correspond to uniform sampling at 10 FPS and 2 FPS, respectively. The middle stance and early swing correspond to sampling at the indicated gait phases (gait percentages are as indicated in Fig. 10).

For BGRU, the patterns were similar but had fewer fluctuations than BMLP because BGRU fused the information from multiple historical frames.

4) Gait-Based Frame Selection: Selecting frames during the middle-stance gait phase showed better performances than other strategies with similar temporal density. Our optimal frame selection strategy reduced the computations by 80% while scarifying only 1%∼6% of the accuracy.

The analysis in Section IV-C3 showed the correlation between the gait phases and uncertainty measures. We further utilized this analysis to select "good" frames for prediction. First, we applied a peak detection algorithm (implemented by MATLAB [33]) to the gyroscope signals to detect the end of the gait cycles. Second, the current gait cycle duration was estimated by averaging the duration of the past five gait cycles. Third, we sampled two subsequent frames based on the estimated gait phases. Two subsequent frames were used to mitigate the impact of observation noises (e.g., motion blur and overexposure). Finally, we trained a BGRU for terrain prediction with only the selected frames. In our experiment, the averaged time for one step was around 1 s (10 frames), and after the frame selection, only 2 frames were processed per step, decreasing around 80% of the computations. We compared the performances of five frame selection settings: uniform sampling at 10 FPS (fps10), uniform sampling at 2 FPS (fps2), selecting 2 frames at the middle stance and the early swing gait phase. Fig. 11 shows that the performance of fps10 (our baseline using all images) was only 1%∼6% better than the performance of middle stance. Middle stance showed around 5%∼8% better performance than early swing. For fps2, the frames were not consistently sampled at the same gait cycle phases, which added difficulty for extracting temporal features, such as walking speed and direction. As a result, fps2 showed the worst performance, especially for larger forecast windows, in which the temporal features were critical.

D. Feasibility for Amputees

Our framework showed high prediction accuracy for training with able-bodied participants and testing with the amputee.

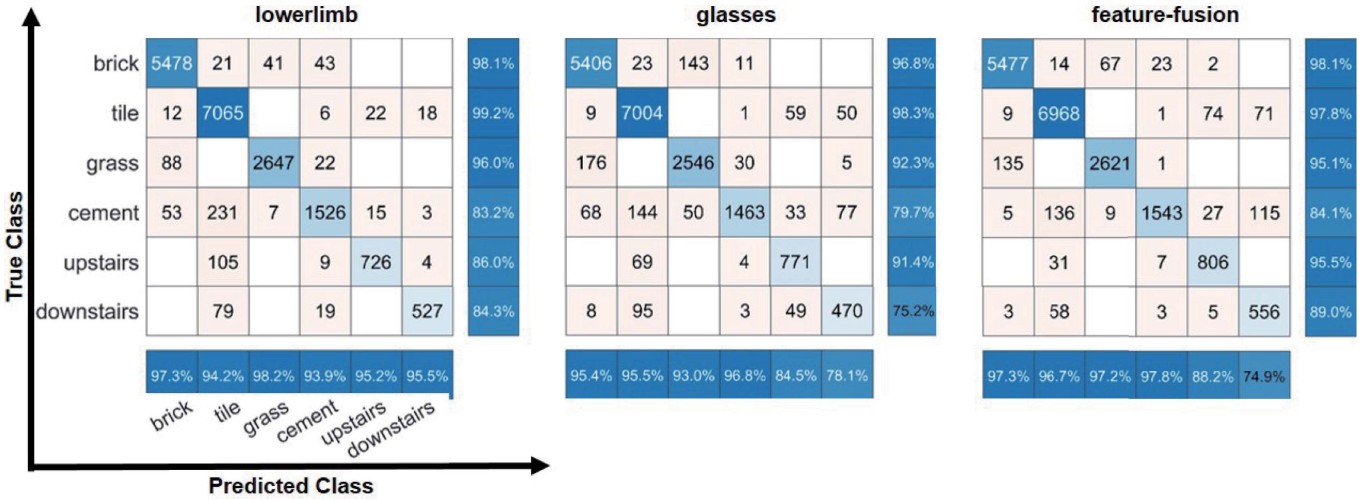


Fig. 12. Confusion matrix of terrain prediction for the amputee with the two data acquisition devices individually and jointly (feature fusion). The results are for predicting the terrain 1 s in the future using the BGRU model. For each confusion matrix plot, the true positive rates and the positive predictive values are in the row and column summery, respectively.

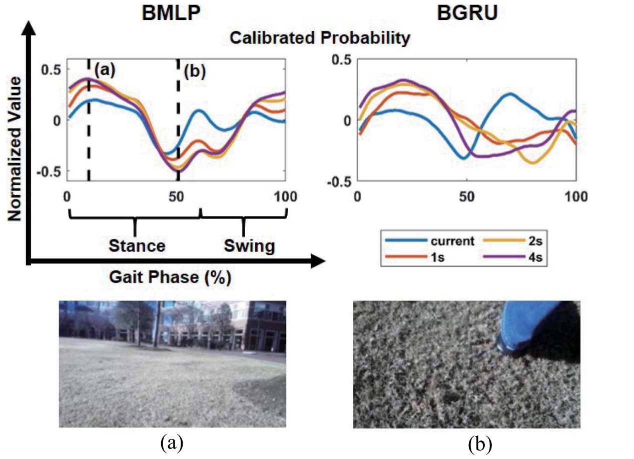
Our analysis in Section IV-C3 also applied to the amputee although the trends were slightly different.

Fig. 12 shows the confusion matrices when training with the able-bodied participants and testing with the amputee. The high prediction accuracy demonstrated the feasibility of training our framework with able-bodied participants and applying the trained framework to amputees. The lower limb showed better performance than glasses for all terrains except upstairs. The overall accuracy was 91.14% for lower limb, 88.96% for glasses, and 93.26% for feature fusion when predicting 1 s into the future using BGRU. It has been observed that amputees tend to look at the terrains near their feet more than able-bodied individuals [34]. Since the framework was trained only with the data of the able-bodied participants, the difference in behavior confused the vision algorithms when estimating the distances to the terrains ahead. Compared with glasses, lower limb would be easier to generalize to the amputee population because the captured videos were only slightly influenced by the difference in gaits. Fig. 13 presents the uncertainty measures aligned with the normalized gait cycle. Figs. 10 and 13 showed similar trends although the highest and lowest uncertainty points shifted due to the different gait patterns for able-bodied participants and amputees. In addition, the results of the amputee showed similar reliability diagrams (see Fig. 7) and uncertainty distributions (see Fig. 9) as the able-bodied participants, indicating the consistency of the uncertainty measures for able-bodied participants and amputees.

E. Feasibility for Mobile Real-Time Computing

The end-to-end inference time was less than 80 ms/frame on a portable embedded system, which is promising for mobile real-time applications.

We implemented our framework with the Tensorflow library. We trained the framework on a PC with an i7-8700K CPU, two NVIDIA 1080Ti GPUs, and 32 GB of RAM. We evaluated our framework on two devices: the PC that we used for training and an NVIDIA Jetson TX2 system [35]. The Jetson TX2 is a portable embedded AI computing system of size 50 mm × 87 mm, weight 85 g, and typical energy usage 7.5 W [36]. We performed the end-to-end inference of our framework for 1000 times with one frame each time (batch size equals to one) and calculated the averaged inference time of each component as well as the entire pipeline. We set the batch size to one in our evaluation to mimic the real-time prediction scenario in practice. Table IV shows the inference time (per frame), the number of floating-point operations (FLOPs), and the number of trainable parameters (Params). We performed 40 iterations of the Monte Carlo dropout sampling for the terrain prediction networks. The reported inference time and FLOPs in Table IV included the computations of all 40 iterations. Since we used MobileNetV2 to extract general image features, the distribution of the parameters in MobileNetV2 can be



(a) (b)

Fig. 13. Top: averaged calibrated probability with respect to the normalized gait cycle for the amputee. The values are Z-score normalized for each gait cycle. Two neural network structures are compared: Bayesian MLP (left) and Bayesian GRU (right). (a) and (b) Two example frames taken by the lower limb camera and the dotted lines indicate the corresponding gait phases.

TABLE IV
INFERENCE TIME AND RESOURCE USAGE

	Jetson TX2 (ms)	Desktop (ms)	FLOPs	Params
Feature Extraction	56.2	13.5	603.7 M	2.2M
Terrain Prediction (BMLP/BGRU)	8.8/12.7	2.2/4.1	74.8/108.8M	0.9/2.2M
Probability Calibration	7.2	0.9	4.5K	2.3K
Entire Pipeline (BMLP/BGRU)	78.4/79.9	18.1/18.9	678.5/712.5M	3.1/4.4M

approximated by deterministic values [37]. Thus, by considering computational efficiency, we did not perform the Monte Carlo sampling for the large feature extraction network (MobileNetV2). On Jetson TX2, the end-to-end inference time of our framework was less than 80 ms, which is potentially efficient for real-time applications. Thus, although we trained each part of the pipeline sequentially, after deploying the end-to-end pipeline, the prediction and uncertainty analysis can be obtained in real time. The algorithm efficiency can be further improved by tuning the number of iterations for Monte Carlo sampling [28] or optimizing inference computing with TensorRT [38]. We left these as our future work.

V. Conclusion and Future Work

In this article, we developed a novel environmental context prediction framework for lower limb prostheses. The framework inherited the advantages of the Bayesian theory and modern deep neural networks. It was able to capture the uncertainty caused by different factors, including observation noise and insufficient or biased training. These uncertainty measures were then projected to one calibrated predicted probability by our probability calibration network. We compared two wearable camera locations and fused them for better performance. Furthermore, we developed a frame selection strategy for the lower limb device inspired by the uncertainty analysis. The results showed promising prediction accuracy and model calibration. Afterward, we demonstrated the feasibility of training our framework with the data of able-bodied participants and applied it to amputees. This study showed the potential for interpreting deep neural networks with uncertainty quantification, utilizing calibrated predicted probability for online decision-making and fusion, and improving the system design by uncertainty analysis. The results in this article may trigger future developments of vision-based context recognition with reliable decision-making, efficient sensor fusion, and improved intelligent system design in multiple applications.

For future work, the proposed vision framework can be integrated into the control systems of wearable robots for real-time evaluation. For example, the control mode for lower limb prosthesis, programed to support amputees in walking on different terrains (e.g., level-ground walking and stair ascent), can be automatically switched according to the terrain predictions. However, pure vision information may not be sufficient to decide the precise switch timing; incorporating other sensing modalities, such as EMG or depth sensors, is needed.

References

[1] (2019). *Tobii Pro Glasses 2 Eye Tracker*. [Online]. Available: https://www.tobiipro.com/product-listing/tobii-pro-glasses-2/

[2] H. A. Varol, F. Sup, and M. Goldfarb, "Multiclass real-time intent recognition of a powered lower limb prosthesis," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 3, pp. 542–551, Mar. 2010.

[3] H. Huang, F. Zhang, L. J. Hargrove, Z. Dou, D. R. Rogers, and K. B. Englehart, "Continuous locomotion-mode identification for prosthetic legs based on neuromuscular–mechanical fusion," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 10, pp. 2867–2875, Oct. 2011.

[4] H. Huang, T. A. Kuiken, and R. D. Lipschutz, "A strategy for identifying locomotion modes using surface electromyography," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 1, pp. 65–73, Jan. 2009.

[5] D. Novak and R. Riener, "A survey of sensor fusion methods in wearable robotics," *Robot. Auto. Syst.*, vol. 73, pp. 155–170, Nov. 2015.

[6] M. Liu, D. Wang, and H. Huang, "Development of an environment-aware locomotion mode recognition system for powered lower limb prostheses," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 4, pp. 434–443, Apr. 2016.

[7] L. Du, F. Zhang, M. Liu, and H. Huang, "Toward design of an environment-aware adaptive locomotion-mode-recognition system," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 10, pp. 2716–2725, Oct. 2012.

[8] N. E. Krausz, T. Lenzi, and L. J. Hargrove, "Depth sensing for improved control of lower limb prostheses," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 11, pp. 2576–2587, Nov. 2015.

[9] Y. Massalin, M. Abdrakhmanova, and H. Atakan Varol, "User-independent intent recognition for lower limb prostheses using depth sensing," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 8, pp. 1759–1770, Aug. 2018.

[10] T. Yan, Y. Sun, T. Liu, C.-H. Cheung, and M. Q.-H. Meng, "A locomotion recognition system using depth images," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 6766–6772.

[11] K. Zhang *et al.*, "Environmental features recognition for lower limb prostheses toward predictive walking," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 3, pp. 465–476, Mar. 2019.

[12] B. Laschowski, W. McNally, A. Wong, and J. McPhee, "Preliminary design of an environment recognition system for controlling robotic lower-limb prostheses and exoskeletons," in *Proc. IEEE 16th Int. Conf. Rehabil. Robot. (ICORR)*, Jun. 2019, pp. 868–873.

[13] J. P. Diaz, R. L. da Silva, B. Zhong, H. H. Huang, and E. Lobaton, "Visual terrain identification and surface inclination estimation for improving human locomotion with a lower-limb prosthetic," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 1817–1820.

[14] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[15] S. Lowry *et al.*, "Visual place recognition: A survey," *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 1–19, Feb. 2016.

[16] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Int. Conf. Mach. Learn.*, 2016, pp. 1050–1059.

[17] Y. Gal and Z. Ghahramani, "Bayesian convolutional neural networks with Bernoulli approximate variational inference," 2015, *arXiv:1506.02158*. [Online]. Available: http://arxiv.org/abs/1506.02158

[18] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1019–1027.

[19] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5574–5584.

[20] J. van der Westhuizen and J. Lasenby, "Bayesian LSTMs in medicine," 2017, *arXiv:1706.01242*. [Online]. Available: http://arxiv.org/abs/1706.01242

[21] R. McAllister *et al.*, "Concrete problems for autonomous vehicle safety: Advantages of Bayesian deep learning," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 4745–4753. [Online]. Available: https://www.ijcai.org/Proceedings/2017/661

[22] S. Singh, C. Arora, and C. V. Jawahar, "First person action recognition using deep learned descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2620–2628.

[23] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[25] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, Jul./Oct. 1948.

[26] N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel, "Bayesian active learning for classification and preference learning," 2011, *arXiv:1112.5745*. [Online]. Available: http://arxiv.org/abs/1112.5745

[27] L. Smith and Y. Gal, "Understanding measures of uncertainty for adversarial example detection," 2018, *arXiv:1803.08533*. [Online]. Available: http://arxiv.org/abs/1803.08533

[28] Y. Gal, "Uncertainty in deep learning," Ph.D. dissertation, Dept. Eng., Univ. Cambridge, Cambridge, U.K., 2016. [Online]. Available: http://mlg.eng.cam.ac.uk/yarin/thesis/thesis.pdf

[29] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[30] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proc. 22nd Int. Conf. Mach. Learn. ICML*, 2005, pp. 625–632.

[31] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using Bayesian binning," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 2901–2907. [Online]. Available:https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4410090/

[32] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 1321–1330.

[33] *MATLAB R2019a*, The MathWorks, Natick, MA, USA, 2019.

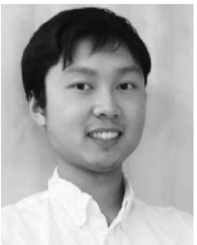
[34] M. Li *et al.*, "Gaze fixation comparisons between amputees and able-bodied individuals in approaching stairs and level-ground transitions: A pilot study," in *Proc. 41st Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2019, pp. 3163–3166.

[35] (2019). *Nvidia Jetson Tx2*. [Online]. Available: https://developer.nvidia.com/embedded/jetson-tx2

[36] (2017). *Nvidia Jetson Tx2 Delivers Twice the Intelligence to the Edge* [Online]. Available: https://devblogs.nvidia.com/jetson-tx2-delivers-twice-intelligence-edge%

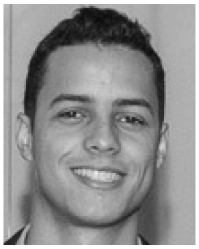
[37] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[38] (2020). *Nvidia Tensorrt, A Programmable Inference Accelerator*. [Online]. Available: https://developer.nvidia.com/tensorrt



Boxuan Zhong received the B.E. degree in electronics and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2015, and the Ph.D. degree in electrical and computer engineering from the North Carolina State University, Raleigh, NC, USA, in 2020.

His current research interests include computer vision, machine learning, and robotics.



Rafael Luiz da Silva (Student Member, IEEE) received the bachelor's degree in technology of industrial automation from the Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, São Paulo, Brazil, in 2008, and the M.Sc. degree in electrical engineering from the Escola Politécnica, University of São Paulo, São Paulo, in 2013. He is currently pursuing the Ph.D. degree in electrical engineering with North Carolina State University, Raleigh, NC, USA.

From 2008 to 2016, he was a Development Analyst with the Continental Automotive Group, Guarulhos, Brazil, working on automotive embedded systems. His research interests include computer vision, machine learning, and signal processing.

Minhan Li received the B.S. and M.S. degrees in mechanical engineering from Tianjin University, Tianjin, China, in 2015 and 2018, respectively. He is currently pursuing the Ph.D. degree in biomedical engineering with the Joint Department of Biomedical Engineering, North Carolina State University, Raleigh, NC, USA, and The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA.

His current research interests include wearable robots, intelligent control, machine learning, and rehabilitation.

He (Helen) Huang (Senior Member, IEEE) received the Ph.D. degree in biomedical engineering from Arizona State University, Tempe, AZ, USA.

She was a Post-Doctoral Fellow in neural engineering with the Rehabilitation Institute of Chicago/Northwestern University, Evanston, IL, USA. She is currently a Professor with the NCSU/UNC Joint Department of Biomedical Engineering and the Director of the Closed-Loop Engineering for Advanced Rehabilitation Core, North Carolina State University, Raleigh, NC, USA, and The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA. Her current research interests include neural-machine interfaces for artificial limbs and exoskeletons, human–robot interaction, adaptive and optimal control of wearable robots, and human movement control.

Dr. Huang is a member of the Society for Neuroscience and the Biomedical Engineering Society. She was a recipient of the Delsys Prize for Innovation in Electromyography, the Mary E. Switzer Fellowship with the National Institute on Disability, Independent Living, and Rehabilitation Research, and the NSF CAREER Award and was named the NC State Faculty Scholar in 2015.

Edgar Lobaton (Member, IEEE) received the Ph.D. degree in electrical engineering and computer sciences from the University of California at Berkeley, Berkeley, CA, USA, in 2009.

He was engaged in research at Alcatel-Lucent Bell Labs, Murray Hill, NJ, USA, in 2005 and 2009. He conducted research at the Department of Computer Science, The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, from 2009 to 2011. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, USA. His research interests include pattern recognition, computer vision, and robotics.

Dr. Lobaton was awarded the 2009 Computer Innovation Fellows Postdoctoral Fellowship.