Kim, K. and Cho, Y., (2020). "Automatic recognition of workers' motions in highway construction by using motion sensors and long short-term memory (LSTM) networks." ASCE Journal of Construction Engineering and Management, 147(3) https://doi.org/10.1061/(ASCE)CO.1943-7862.0002001.

Automatic recognition of workers' motions in highway construction by using motion sensors and long short-term memory (LSTM) networks

Kinam Kim[1] and Yong K. Cho[2]

1) Graduate Student, School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA, USA; PH: 1-404-398-5446; E-mail: kkim734@gatech.edu

2) Associate Professor, School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA, USA; PH: 1-402-385-2038; E-mail: yong.cho@ce.gatech.edu (corresponding author)

Abstract: Monitoring and understanding construction workers' behavior and working conditions are essential to achieve success in construction projects. The dynamic nature of construction sites has heightened the awareness of the need for improved monitoring of the individual workers on the sites. Although several studies have shown promising results in automated motion and activity recognition using wearable motion sensors, their technical and practical feasibility was not properly validated in actual jobsites. Motion recognition models have to be evaluated in actual conditions because the motion sensor data collected in controlled conditions, and actual conditions can have different characteristics. This study proposes Long Short-Term Memory (LSTM) networks for recognizing construction workers' motions. The LSTM networks were validated through case studies in one bridge construction site and two road pavement sites. The LSTM networks showed classification accuracies of 97.6%, 95.93%, and 97.36% from three different field test sites, respectively. Through the case studies, the technical and practical feasibility of the LSTM networks was properly investigated. With the LSTM networks, it is expected that the individual workers' behavior and working conditions

can be automatically monitored and managed without excessive manual observation.

Keywords: Construction worker; Motion recognition; Monitoring, Deep learning; Long short-term memory

Introduction

Construction tasks involve various activities composed of one or more body motions. It is essential to understand the dynamically changing activities and motions of construction workers to effectively manage the workers for improving safety and productivity. Because construction projects are inherently labor-intensive and rely heavily on manual tasks, the understanding of activities and motions of individual workers is required to ensure and improve their safety and productivity.

According to the Construction Chart Book from the Center for Construction Research and Training (CPWR 2018), the rate of Work-related Musculoskeletal Disorders (WMSDs) in the construction industry in 2015 was 34.6 per 10,000 Full-Time Equivalent (FTE) workers. This was 16% higher than the rate in all industries (29.8 per 10,000 FTEs). WMSDs are work-related injuries of the muscles, joints, tendons, and nerve tissues (CPWR 2018). These injuries often result from posture-related safety risks and are frequently observed in construction workers because construction tasks require repetitive movement, high force exertion, vibration, and awkward postures, all of which can cause WMSDs (NIOSH 2019).

Meanwhile, the productivity of the construction industry has hardly improved or even declined since 1995, while the productivity of other industries has been improved noticeably (McKinsey & Company 2015). Because the construction industry is labor-intensive, labor productivity directly affects construction productivity (Ghate et al. 2016). While the growth of labor productivity in the overall global industry has been 2.8% since 1995, growth in global construction industries has been only 1% (McKinsey Global Institute 2017).

To address concerns about safety and productivity, there have been several efforts to mitigate safety risks and improve productivity by training, educating, and manually monitoring

workers. However, these passive and manual methods are labor-intensive and error-prone and cannot be used to collect holistic data from individual workers. Hence, it is important to identify an approach to better monitor workers and collect relevant data with regard to safety and productivity at an individual level.

To facilitate monitoring and managing individual workers, motion recognition methods have been widely utilized. While several efforts have been introduced to utilize motion recognition methods in construction projects, they have not been deployed and evaluated with actual workers under actual jobsite conditions. Because motion patterns vary depending on subjective motion biases and types of given tasks, it is important to validate a recognition method through actual field implementation. This study proposes an automated motion recognition model for construction workers using Long Short-Term Memory (LSTM) networks (Kim and Cho 2020) that are designed to learn sequential information. A two-stacked LSTM network recognizes the workers' motions from motion sensor data from two Inertial Measurement Units (IMUs) attached to each worker by processing sequences of the motion data. To validate the model, in-depth case studies with three different construction sites have been conducted. Each case generated a separate motion dataset to be used for training an LSTM network. The three case studies provide an opportunity to evaluate the technical and practical feasibility of the model. With the motion recognition model, various motions of workers can be properly recognized from actual construction tasks and utilized as primary elements for managing the workers.

Automated motion recognition with motion sensor data

Motion recognition refers to a pattern-based method of recognizing human motion

states using sensors, such as accelerometers, gyroscopes, and cameras (Pei et al. 2015). In construction projects, motion recognition has gained much attention because of its capability to identify workers' working conditions about safety and productivity without excessive observation.

In regard to safety, motion recognition can be used to identify unsafe postures of workers. In one study, awkward postures were identified by using a Support Vector Machine (SVM) classifier with a supervised motion tensor decomposition to process data from wearable IMUs (Chen et al. 2017). This enabled the classifier to be efficiently implemented in terms of computational capacity. A binary classifier recognized near-miss falls by using one-class SVM with motion data from wearable IMUs (Yang et al. 2016). And insole pressure sensors were used to detect workers' loss of balance, which can cause fall accidents (Antwi-Afari et al. 2018). Changes of pressure on workers' feet were utilized as a clue to identify unsafe motions. In this study, five types of machine learning algorithms were implemented: Artificial Neural Network (ANN), Decision Tree (DT), Random Forest (RF), k-Nearest Neighbor (k-NN), and SVM. Similarly, IMUs attached to the ankles have been used to analyze gait stability, which can be an indicator of fall risks (Jebelli et al. 2015; Yang et al. 2019; Yang and Ahn 2019). A Convolutional Neural Network (CNN) was integrated with LSTM for construction worker's motion recognition using five motion sensors and tested under the controlled environment (Zhao and Obonyo 2019). The developed model in this study recognized the motions that can cause musculoskeletal disorders such as bending, squatting, and kneeling.

Motion recognition can be also used to calculate workers' productive time and analyze their productivity (Akhavian and Behzadan 2016a; Nath and Behzadan 2017). Productive time was calculated by counting idle time, as classified by a machine learning classifier. In these

studies, a smartphone with an embedded IMU was used to collect motion sensor data. A SVM classifier was developed to categorize masonry workers into the expert and inexpert groups based on their motions for comparing the productivity (Alwasel et al. 2017).

Also, motion recognition methods were utilized to identify various motions and activities using machine learning algorithms (Kim and Cho 2020; Ryu et al. 2016, 2019; Su et al. 2014; Valero et al. 2017). Five types of machine learning algorithms were implemented to recognize four activities of construction workers (Akhavian and Behzadan 2016b). A smartphone was attached to the arm to collect motion data. Likewise, an accelerometer-embedded wristband sensor was utilized to recognize the activities of a masonry worker by using machine learning algorithms (Ryu et al. 2016, 2019). These studies investigated the impacts of window sizes used in pre-processing on the classification performance. A two-stacked LSTM network based on the effective quantity and locations of motion sensors was developed for recognizing various motions of the workers (Kim and Cho 2020). While the above-mentioned studies used machine learning algorithms to classify motion data into motions and activities, a wireless motion sensor network has also been used to analyze workers' motions without machine learning algorithms (Valero et al. 2017; Yan et al. 2017).

As described above, motion recognition methods have been widely utilized in construction projects. Although several efforts showed promising results within a controlled environment, the existing approaches were not practically validated through actual field experiments with actual construction workers. Such validation is essential because data collected under controlled conditions does not reflect the dynamic nature of jobsites and workers. In addition, subjects who participated in controlled working environments were generally asked to perform the task following certain guidelines, e.g., performing specific

motions with well-distributed counts. However, under actual jobsite conditions, workers' motions tend to be unpredictable and have imbalanced distribution across various motions. These features can be an obstacle to implementing a classification model in real-world conditions if it is developed under controlled conditions. To address these issues, this study proposed an LSTM network, one of the deep-learning algorithms designed to learn sequential information, to recognize various motions of construction workers and validated the network through three different case studies at actual construction sites.

There have been several studies on the vision-based motion and activity recognition approach. These studies are not considered in the literature review of this paper because this study focuses on validating a deep learning-based motion recognition method using motion sensors in real construction sites. Due to practical issues, it may not be possible to validate vision-based approaches in real construction sites. First of all, a vision-based approach is not robust to occlusion due to a camera's line of sight. Second, a vision-based approach is sensitive to lighting conditions that affect recognition performance. Third, a privacy issue can be raised when cameras are used on construction sites, because they collect personal information (e.g., faces) in addition to motions. Last but not least, the maintenance of multiple cameras at a construction site is challenging because of power supply issues and the frequent need to relocate the cameras. Hence, this study focuses on a motion recognition method using motion sensors.

Methodology

This study proposes an LSTM network for workers' motion recognition validated through three different real-world cases. The LSTM network is implemented in three steps: (1)

dataset generation; (2) LSTM network implementation; and (3) performance evaluation. Fig. 1 illustrates the implementation process of the proposed model.

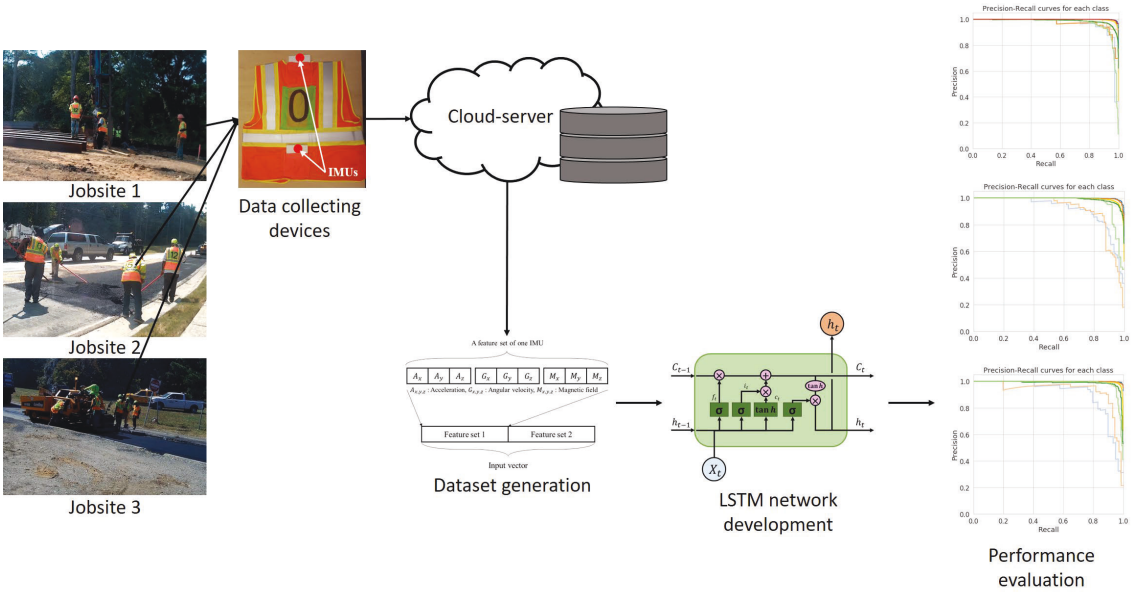


Fig. 1. A development process of the proposed model.

Dataset generation

The three different motion datasets are generated from three jobsites. Each dataset contains motion sensor data from two IMUs carried by each worker—one on the back of the neck and one on the lower back. These two locations are selected to reflect the movements of each worker's upper and lower body. It was found that motion recognition methods with two IMUs located a certain distance apart, such as neck and hip or head and hip, showed similar performance compared to methods using 17 IMUs located throughout the entire body (Kim et al. 2019; Kim and Cho 2020). This indicates that given correct placement and adequate spacing, data collected using two IMUs are sufficient to reflect the entire body's movement.

Each IMU generates a feature set composed of 9 values: acceleration (3 values for x,

y, and z axes), angular velocity (3 values for x, y, and z axes), and magnetic field (3 values for x, y, z axes), as shown in Fig. 2. These values are raw data measured from the accelerometer, gyroscope, and magnetometer embedded in the IMU, respectively. Then, two feature sets from two IMUs are concatenated to form an input vector. One input vector includes 18 values. Once input vectors are generated, each vector is labeled as a particular class by comparing it with recorded videos from the jobsites.

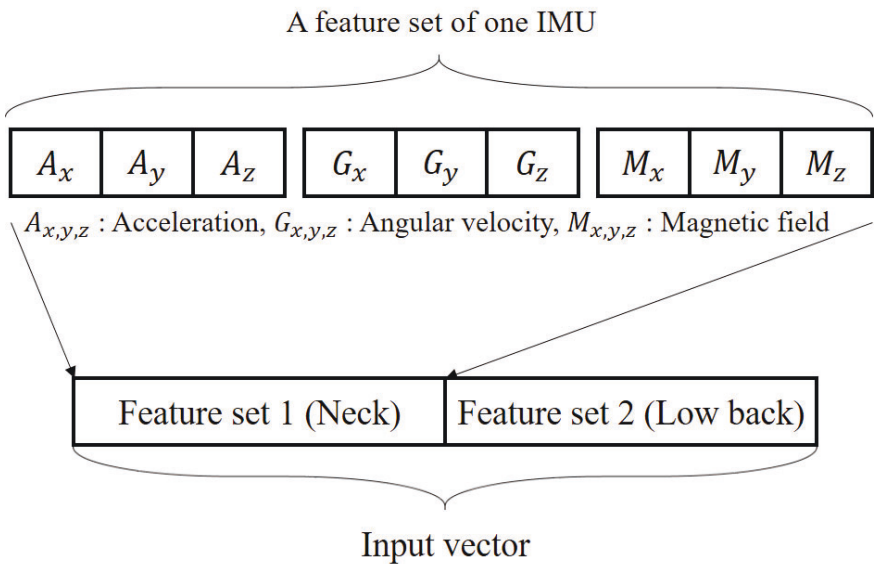


Fig. 2. Input vector generation.

There are 14 possible motion classes: standing, bending-up, bending, bending-down, squatting-up, squatting, squatting-down, walking, twisting, working overhead, kneeling-up, kneeling, kneeling-down, and using stairs. These motion classes are theoretical classes; actual workers may use only some of them, depending on their given jobs. Among the 14 motions, 6 are defined as transitioning motions: bending-up, bending-down, squatting-up, squatting-down,

kneeling-up, and kneeling-down. These motions are derived from the base motions of bending, squatting, and kneeling to reduce the loss of information. For example, bending-up and bending-down are transitioning motions from the bending motion to other motions and vice versa.

Long Short-Term Memory (LSTM) network implementation

Once the three datasets are generated, a two-stacked LSTM network is implemented. An LSTM network is a recurrent neural network designed to learn sequential information using memory cells that store and output information, facilitating the learning of temporal relationships on long-time scales (Ordóñez and Roggen 2016). While conventional machine learning classification algorithms categorize each input datum to a class independently, an LSTM network classifies a sequence of input data to a class. This is an important characteristic that can be useful in motion recognition because a particular motion can be interpreted as a result of a sort of motion. For example, a bending motion is taken after a sequence of stooping motions from standing or walking motions. Existing approaches with conventional machine learning algorithms have utilized feature extraction techniques with segmented data using sliding windows to reflect this temporal characteristic to the classification model. However, the order of motions in a sequence is not effectively reflected with conventional feature extraction techniques. Furthermore, the selection of feature types can significantly affect the performance of the classification models. In this sense, an LSTM network can be a solution to reflect the temporal characteristic because it learns sequential information. An LSTM network uses the gating concept—i.e., a mechanism based on pointwise multiplication operations and activation functions. Using the gating concept, the information that passes the gate is selectively added to

or removed from the memory cell. Fig. 3 illustrates the basic structure of the LSTM network.

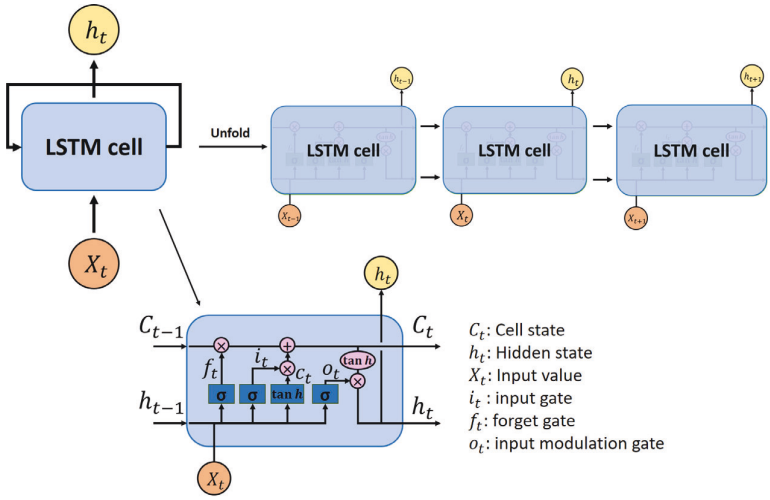


Fig. 3. A basic structure of the LSTM network.

In the LSTM cell, the information passes the gates and updates the states as follows. First, input values x_i and the previous hidden state h_{t-1} pass through the forget gate f_t. The forget gate outputs a value between 0 (complete removal of the information) and 1 (complete retention of the information). Second, the input values x_i and the previous hidden state h_{t-1} pass through input gate i_t to store new information in the new cell state C_t. These values also pass through the input modulation gate $\tilde{C}_t$ with a hyperbolic tangent activation function so that the output value ranges between -1 and 1, which reflects the amount of the information to be forgotten. Then, the old cell state C_{t-1} is updated into the new cell state C_t by multiplying the old cell state and the forget gate's output, and by adding the multiplication of the output values of the input gate and input modulation gate. Subsequently, input values and the previous hidden state, x_i and h_{t-1}, pass through the output gate with a sigmoid activation function to determine the parts of the cell state that will be the output. Lastly, the cell state C_t passes through a hyperbolic tangent function. This is multiplied by the output of the output gate to

calculate the new hidden state h_t. The equations of the gates and states are as follows:

$$\begin{cases} i_t = \sigma(W_{xi}x_t + V_{hi}h_{t-1} + b_i) \\ f_t = \sigma(W_{xf}x_t + V_{hf}h_{t-1} + b_f) \\ o_t = \sigma(W_{xo}x_t + V_{ho}h_{t-1} + b_o) \\ \tilde{C}_t = \tanh(W_{xc}x_t + V_{hc}h_{t-1} + b_c) \\ C_t = f_t \otimes C_{t-1} + i_t \otimes \tilde{C}_t \\ h_t = o_t \otimes \tanh(C_t) \end{cases} \quad (1)$$

In the equations, σ is the sigmoid function defined as $\sigma(x) = (1 + e^{-x})^{-1}$. i_t, f_t, o_t, $\tilde{C}$, C_t, and h_t are the outputs of the input gate, forget gate, output gate, input modulation gate, cell state, and hidden state at time t, respectively. $\otimes$ is a pointwise multiplication operator. W_{xi}, W_{xf}, W_{xo}, W_{xc}, V_{hi}, V_{hf}, V_{ho}, and V_{hc} are the coefficient matrix. b_i, b_f, b_o, and b_c are bias vectors. Here, the coefficient matrix and bias vectors are learnable parameters. By updating these parameters, the network learns the amount of information that passes through the LSTM cell.

The two-stacked LSTM network structure developed in (Kim and Cho 2020) is adopted in this study. In the network, two LSTM cells are connected to each other to make the network deeper. Fig. 4 illustrates the structure of the LSTM network. To generate input sequences, input vectors are segmented into sequences with a certain length. The length is a parameter to be determined by hyper-parameter tuning. In the network, the input sequences are fed into a fully connected layer, followed by a Rectified Linear Unit (ReLU). Commonly used because it outperforms a sigmoid function, the ReLU layer is implemented to improve the performance of the network (Zhao et al. 2018). The fully connected layer used prior to the first LSTM cell is utilized to make the network deeper and allow it to learn the characteristics of the motion data other than sequential information. For instance, as the data contains 18 features including 2 sets of triaxial acceleration, gyroscope, and magnetic field, some features can be

correlated with each other. The fully connected layer is added between an input layer and the first LSTM cell to deal with such characteristics based on empirical knowledge. To regularize the network and avoid overfitting, a dropout technique is implemented in the second LSTM cell. The dropout technique probabilistically excludes the recurrent components such as input, output, and hidden state in the update process. After the second LSTM cell, the last output of the input sequences is fed into the fully connected layer. This is because the target motion to be classified is at the end of the motion sequences. Finally, the output of the fully connected layer is fed into the softmax layer to convert class scores into probabilities so that the motion with the highest probability can be identified.

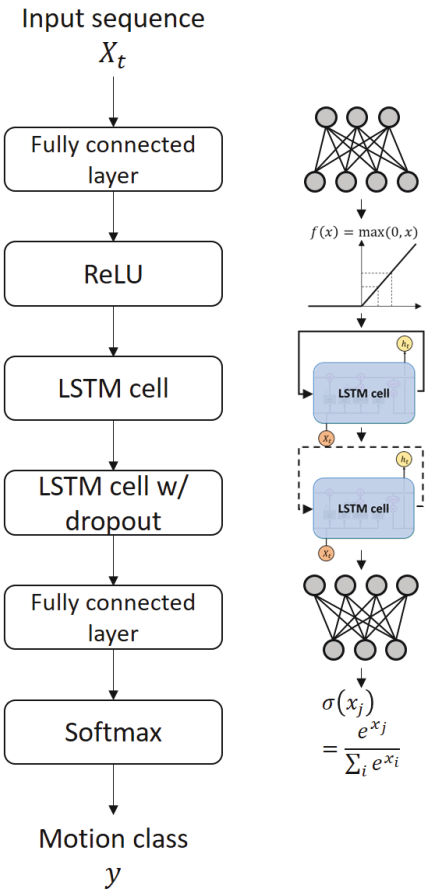


Fig. 4. The structure of the proposed two-stacked LSTM network.

Performance evaluation

In the training procedure, each dataset from three jobsites is split into three subsets: training, validation, and testing. The training set is for fitting the network, i.e., coefficient matrix and bias vectors. The validation set is for evaluating the network with unbiased data during the hyper-parameter tuning. The testing set is for evaluating the network with unseen data—i.e., data not used in either network fitting or hyper-parameter tuning. Once the hyper-parameter tuning is done, confusion matrix and precision-recall curves are utilized to evaluate the performance. Through the performance evaluation, the networks are properly validated. This validation process is repeated three times with three different datasets collected from real construction projects so that the network can be evaluated in terms of technical and practical feasibility.

Experiments and results

Dataset generation

This study includes case studies of three different construction sites. Fig. 5 shows three jobsite scenes. The first site was a bridge construction site. Four workers participated in the study. The tasks given to the workers were related to pile installation. The second site was an asphalt road paving site. Three workers participated in the study. The tasks given to the workers involved asphalt spreading. The third site was another asphalt road paving site. Four workers participated in the study, and their tasks also involved asphalt spreading.

To collect motion sensor data from the workers, data collecting devices developed by Robotics and Intelligent Construction Automation Laboratory (RICAL) group at Georgia Institute of Technology were utilized. The devices were carried by workers wearing safety vests

with pockets on the back of the neck and lower back, as shown in Fig. 6. The devices are equipped with a wireless communication module for Wi-Fi, a micro processing unit, data storage, battery, and an IMU. The IMU used in this study consisted of three triaxial sensors, including accelerometer, gyroscope, and magnetometer, which have digital resolutions of 0.98 mg, 0.004°/s, and 0.3 μT, respectively. Using these features, IMU data can be automatically collected, uploaded, and stored on a cloud server.

Fig. 5. Jobsite scenes; (a) jobsite 1, (b) jobsite 2, and (c) jobsite, motion examples; (d) standing, (e) walking, (f) bending-up, (g) bending, (h) bending-down, (i) twisting, and (j) working overhead.

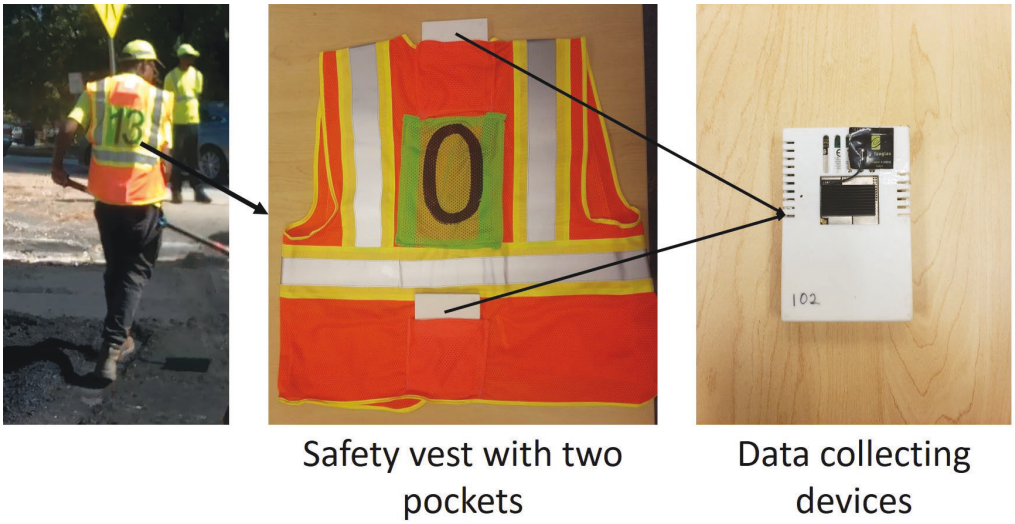


Fig. 6. Data collecting devices and a safety vest with two pockets.

Three datasets were collected from the workers in the three jobsites. The first, second, and third datasets contain 32,959 data points, 9,577 data points, and 10,743 data points, respectively. The collected data points were labeled with timestamps to be manually compared with the recorded videos later. IMU data from two devices were concatenated to form input vectors and normalized to have a unit norm. Then, 6 input vectors were segmented into a sequence which was used as an input for the LSTM network. Because the IMU in the data collecting device measures 30 data points in a second, one sequence contains the data collected every 0.2 seconds. Adjacent sequences are overlapped with one data point, which means the overlap ratio is 16.6% (1/6). As a result of segmentation, the dimensions of the first, second, and third datasets in a sequential form are 32,954 × 18 × 6; 9572 × 18 × 6; and 10,738 × 18 × 6.

Long Short-Term Memory (LSTM) network implementation

Three LSTM networks were implemented using Tensorflow, which is an artificial

intelligence library using data flow graphs to build models. Each network was independently implemented with the dataset from each jobsite. A computer equipped with Intel® Core™ i7-8650U CPU, Intel® UHD Graphics 620, and 16 GB RAM was used to implement the networks. The sequential datasets were shuffled and split into training data and test data that occupy 70% and 30% of the entire dataset, respectively. Then, the training data were split again into training data and validation data that occupy 70% and 30% of the original training data, respectively.

Once the networks were fit for each set of training data, the hyper-parameters were tuned by a grid search and a minor random search. Using the grid search approach, the best parameter set that showed the highest accuracy was selected and used in the minor random search. By randomly adjusting the parameters slightly, under- and overfitting issues were resolved. The final hyper-parameters for each network are shown in Table 1. The losses and accuracies over iteration with the hyper-parameters are shown in Figs. 7 through 12. The losses and accuracies indicate the cross-entropy of the result after the softmax function is applied and the ratio of the number of the correctly classified samples to the number of the entire samples. For the LSTM network with the first dataset, the losses over iteration graphs showed that the losses were converged and small enough after 300 epochs. For the LSTM networks with the second and third datasets, the same phenomenon was observed after 700 epochs and 350 epochs, respectively. It was found that the differences between training losses and validation losses of the three LSTM networks were small enough, which means the networks were well-trained without overfitting. An overfitted network too closely fits to the training data and shows low performance on the testing data. To avoid overfitting, l_2 norm regularization and a dropout technique were implemented. Dropout probability was set to 0.4 or 0.5 only in the training process. This means that the recurrent connections in the LSTM cells were excluded with 40%

or 50% of probability. Adam optimizer (Kingma and Lei Ba 2015) was utilized to minimize a loss function, which is the cross-entropy of the result after the softmax function is applied.

Table 1. Hyper-parameters of the LSTM networks.

Hyper-parameter	Value		
	LSTM 1 (Jobsite 1)	LSTM 2 (Jobsite 2)	LSTM 3 (Jobsite 3)
The number of hidden units	180	180	120
L2 regularization factor	0.0002	0.0004	0.0002
Learning rate	0.0005	0.0005	0.001
The number of epochs	400	800	400
Batch size	128	128	128
Dropout probability	0.4	0.5	0.5

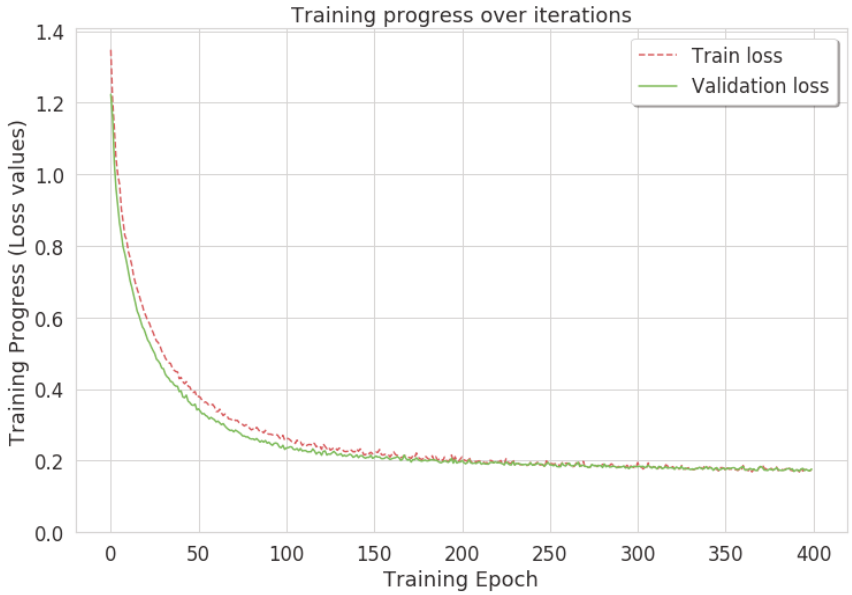


Fig. 7. Training and validation losses over iteration of the LSTM 1.

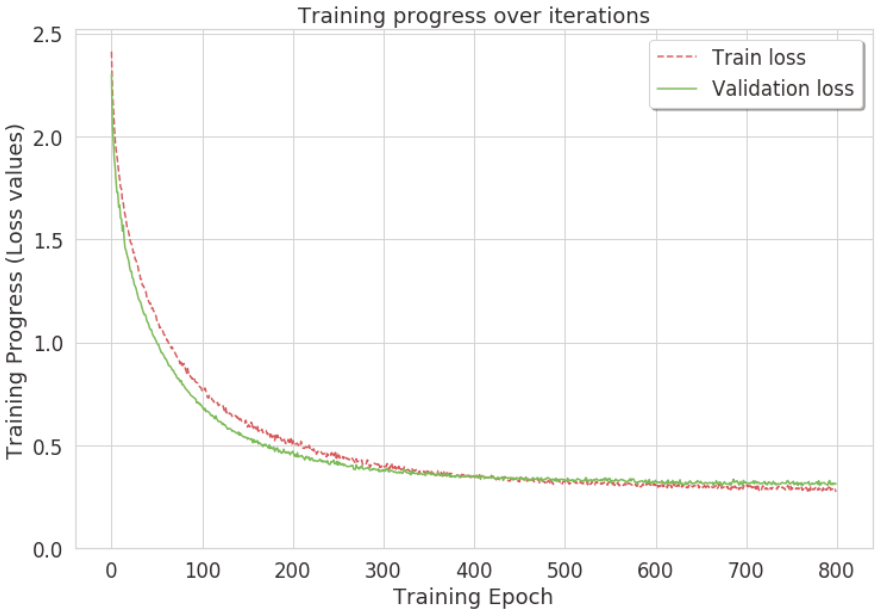


Fig. 8. Training and validation losses over iteration of the LSTM 2.

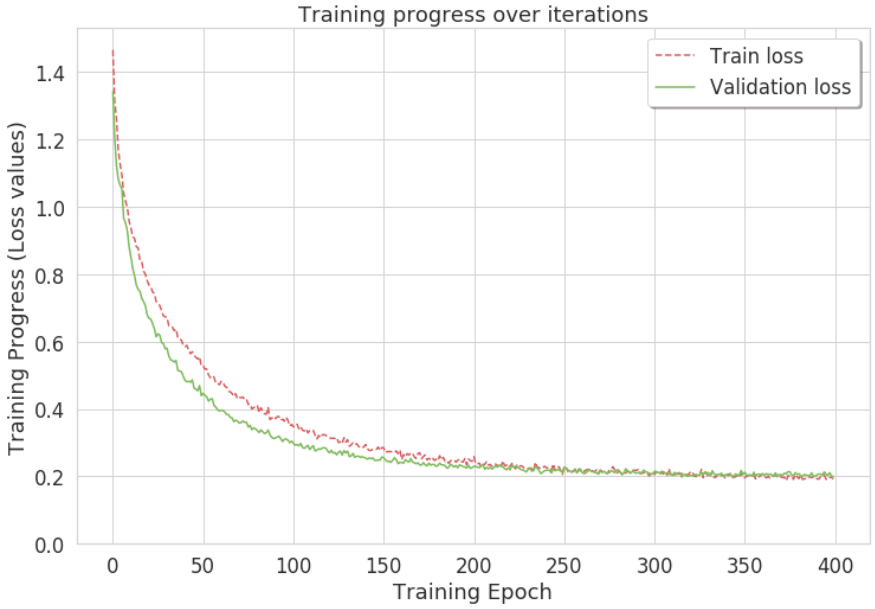


Fig. 9. Training and validation losses over iteration of the LSTM 3.

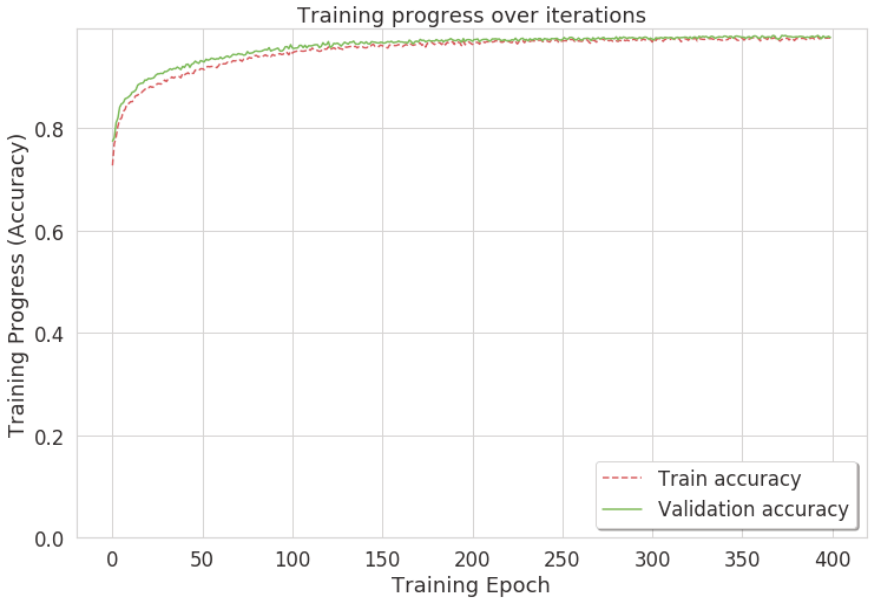


Fig. 10. Training and validation accuracies over iteration of the LSTM 1.

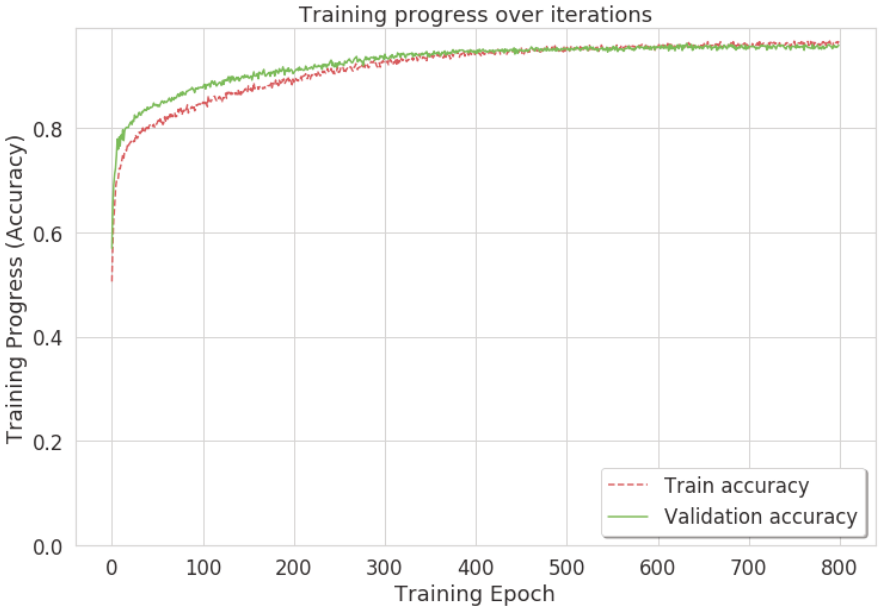


Fig. 11. Training and validation accuracies over iteration of the LSTM 2.

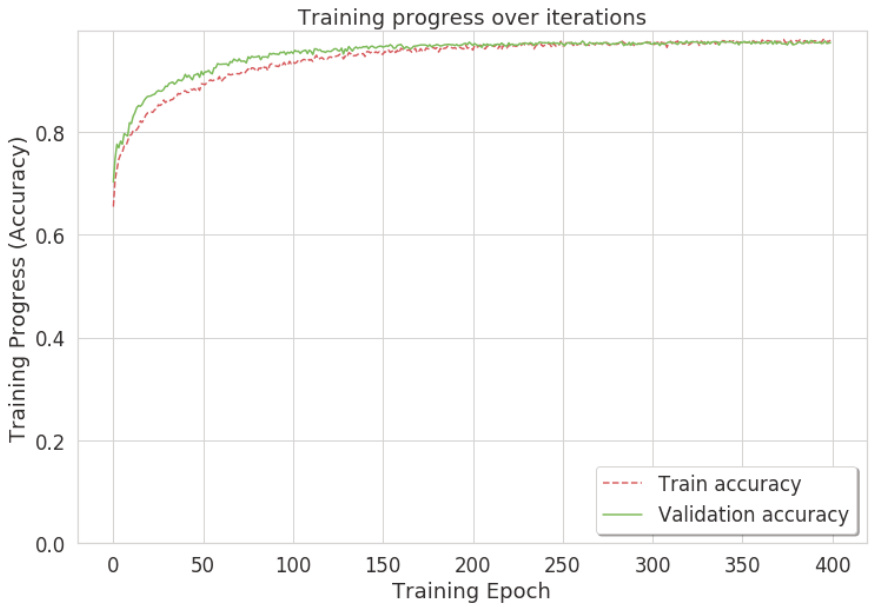
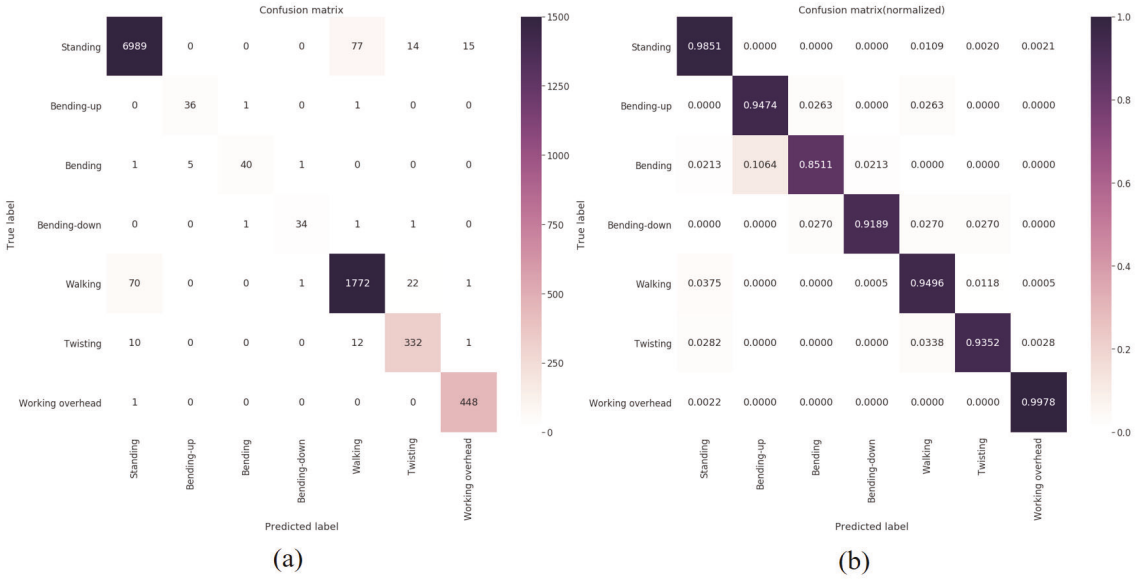


Fig. 12. Training and validation accuracies over iteration of the LSTM 3.

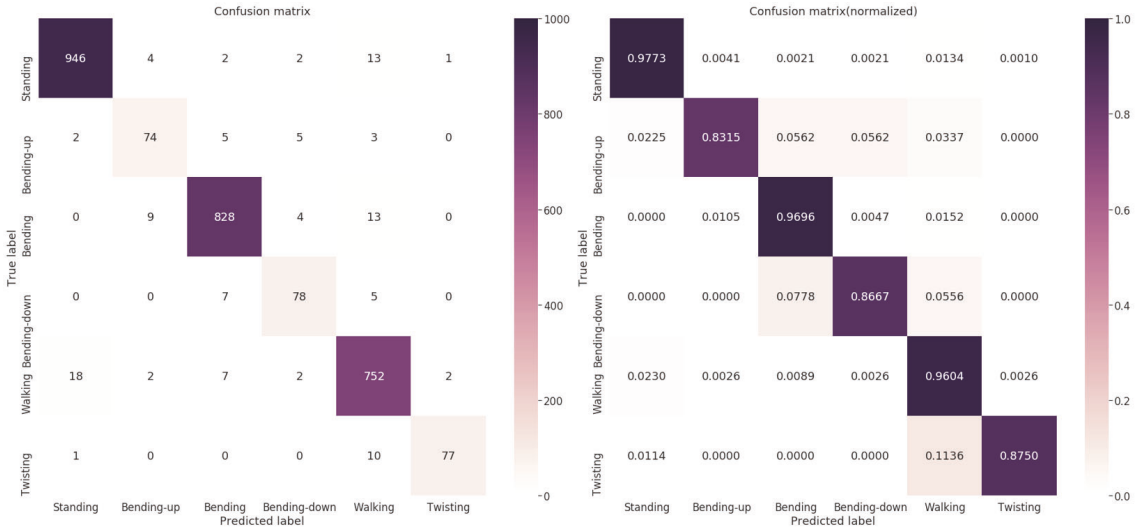
Performance evaluation

As a result, the three LSTM networks showed accuracies of 97.6%, 95.93%, and 97.36% on the testing data, respectively. Confusion matrixes with normalization and without normalization are shown in Figs. 13 through 15. In the confusion matrixes, diagonal values are the counts of the samples that are correctly classified, and off-diagonal values are the counts of the samples that are incorrectly classified. The color-coded elements of the matrixes without normalization represent the absolute counts of each classification, and the color-coded elements of the matrixes with normalization represent a portion of each classification among the ground truth. Since the collected datasets are imbalanced, the normalized confusion matrixes represent the results more effectively.



(a) (b)

Fig. 13. Confusion matrixes of the LSTM 1; (a) without normalization and (b) with normalization.



(a) (b)

Fig. 14. Confusion matrixes of the LSTM 2; (a) without normalization and (b) with normalization.

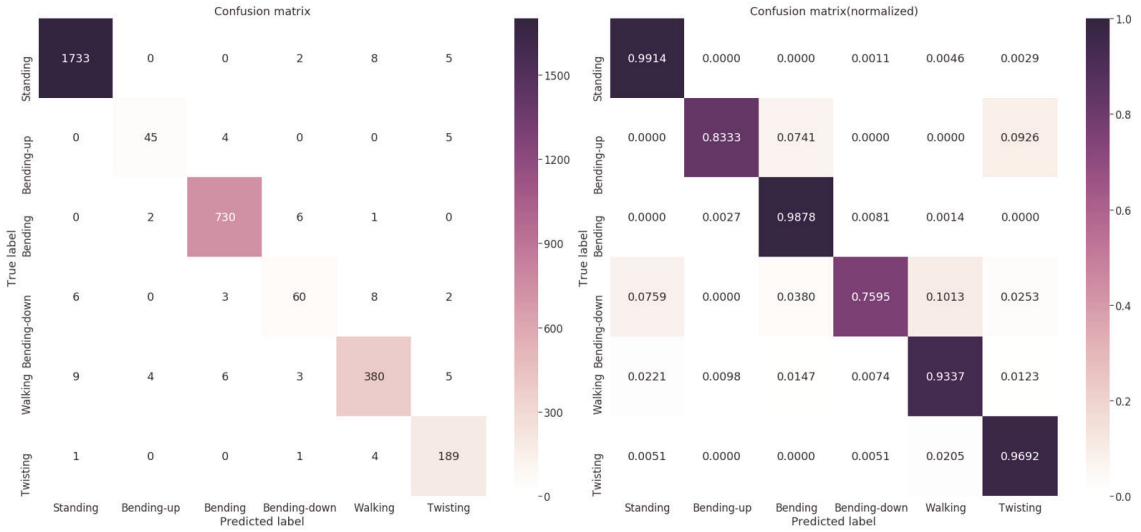


Fig. 15. Confusion matrixes of the LSTM 3; (a) without normalization and (b) with normalization.

To evaluate the networks with regard to the imbalanced datasets, Precision-Recall (PR) curves for each class were derived as shown in Figs. 16 through 18. The PR curves are evaluation measures for the classification that allows the visualization of the performance of the classifier at a range of thresholds (Boyd et al. 2013). The PR curves are used to evaluate binary classification models trained with an imbalanced dataset. In the imbalanced dataset, some classes occupy a larger portion of the dataset than the other classes. Since the datasets used in the study are imbalanced with multi-classes, the PR curves for each class are used in the evaluation.

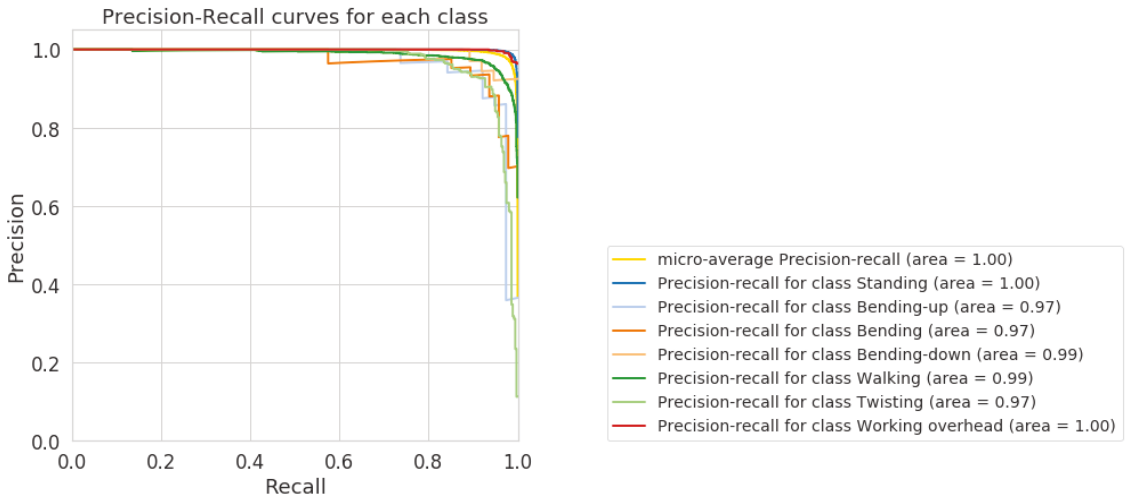


Fig. 16. Precision-recall curves of the LSTM 1.

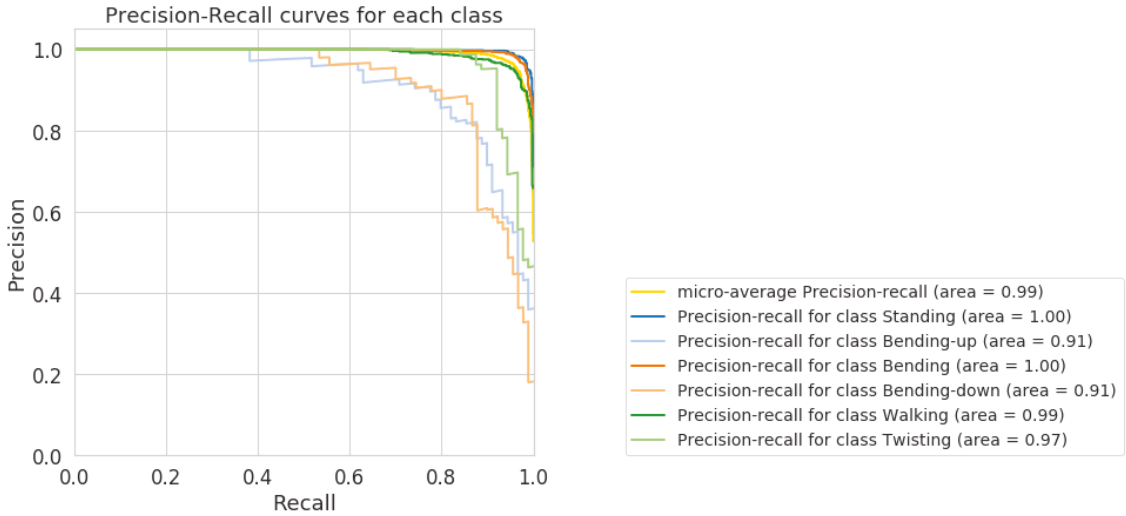


Fig. 17. Precision-recall curves of the LSTM 2.

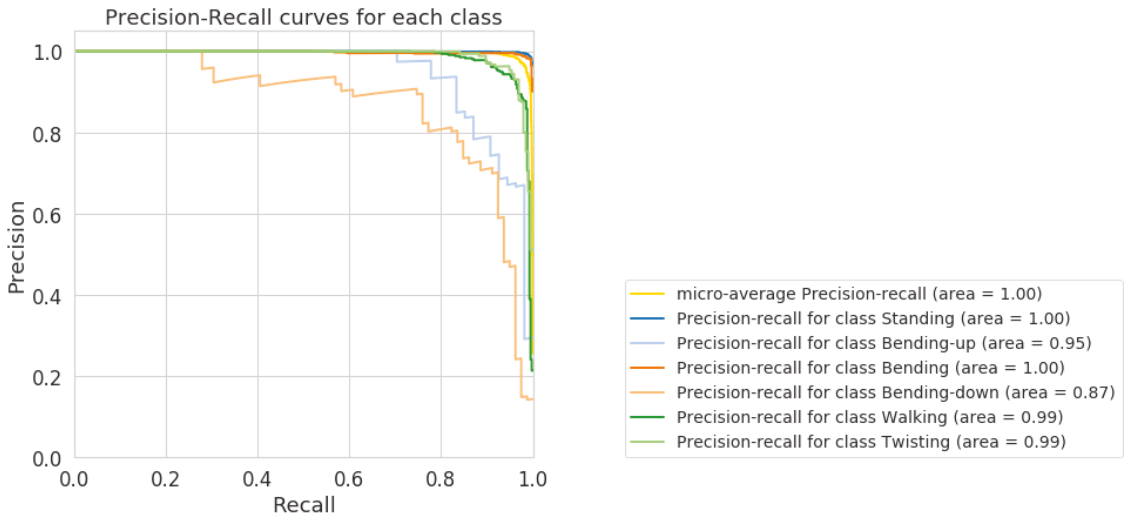


Fig. 18. Precision-recall curves of the LSTM 3.

The PR curves are interpreted by calculating the areas under the curves. The area ranges from 0 to 1, where 0 indicates that the classifier completely failed to classify the data, and 1 indicates that the classifier completely classified the data. In addition to the area, the classifier can be evaluated as a better classifier when the curve is close to the right upper corner. The legends of Figs. 16 through 18 show the areas of the curves. For the first LSTM network, all areas were over 0.97, which means that the network properly classified the data by considering the imbalance of the dataset. For the second and third LSTM networks, areas of primary motion classes such as standing, bending, walking, and twisting were over 0.97, while areas of transitioning motion classes such as bending-up and bending-down were 0.85, 0.91, and 0.95.

Discussion

In the proposed study, the two-stacked LSTM networks demonstrated accuracies of 97.6%, 95.93%, and 97.36% on the testing data, respectively. Compared to the existing other

methods, the networks showed significantly improved performance in terms of the number of sensors, the number of classes, and accuracy, as shown in Table 2. The improved performance of the networks was achieved due to the capability of the two-stacked LSTM cells to learn sequential information and associated layers utilized in the networks such as the fully connected layer and the ReLU layer prior to LSTM cells. These components allowed the networks to learn a time-dependent characteristic of motions effectively. Also, the datasets used in this study were collected from actual workers in actual jobsites, while the existing methods collected data from actual workers (or non-workers) in a controlled environment. This indicates that the implemented networks were able to learn the characteristic of the workers in actual construction sites. The networks were validated through tests in the three actual construction sites. Among the 14 possible motion classes, some were not observed in the classification results. In the case of the first jobsite, 7 motions—squatting, squatting-up, squatting-down, kneeling, kneeling-up, kneeling-down, and using stairs—were not observed. This indicates that the dataset reflected only the characteristics of the given task in the first jobsite, which was pile installation. To be specific, the workers' roles were to support the crane, which moved the piles and dropped a hammer for pile driving. They were mainly working on holding wires to place the piles in the pile driver. Hence, most of the motions were standing, walking, twisting, and working overhead.

In the cases of the second and third jobsites, 8 motions—the 7 unobserved motions from the first jobsite, as well as working overhead—were not observed. The given tasks in the second and third jobsites were spreading and flattening road pavement materials. Workers were mainly working on pushing and pulling an asphalt lute. Hence, standing, bending, and walking motions occupied most of the datasets. These distributions inevitably result in imbalanced

datasets. This imbalance allows the motions involved in the particular tasks to be identified. The identified types of motions involved in different tasks and their distributions can be used as fundamental elements for worker's behavior analysis.

Table 2. Comparison of the performance with existing methods.

Method	Data collection environment, Participant	# of sensors	# of classes	Accuracy
k-NN [Akhavian and Behzadan 2016b]	Controlled Environment, Non-workers	1	5	79.79%
Multi-class SVM [Ryu et al. 2019](Ryu et al. 2019)(Ryu et al. 2019)	Controlled Environment, Actual workers	1	4	88.10%
Convolutional LSTM [Zhao and Obonyo 2019]	Controlled Environment, Actual workers	5	8	85.20%
The LSTM 1 in this study	Actual jobsite, Actual workers	2	7	97.60%
The LSTM 2 in this study	Actual jobsite, Actual workers	2	6	95.93%
The LSTM 3 in this study	Actual jobsite, Actual workers	2	6	97.36%

Theoretically, networks developed with evenly distributed datasets are expected to show the best classification performance. However, datasets collected from actual construction workers will undoubtedly be imbalanced, as shown in the result of this study. Thus, any performance evaluation needs to account for an imbalanced dataset. In this study, PR curves were utilized to evaluate the networks. As a result, the areas under the curves of the primary motions, such as standing, bending, walking, twisting, and working overhead, were over 0.97. This indicates that the primary motions were properly recognized. On the other hand, the areas under the curves of the transitioning motions, such as bending-up and bending-down, were

smaller than those under the curves of primary motions. This is because the number of data for transitioning motions is much smaller than for primary motions. Since the LSTM networks learn the information from the data, fewer data result in the networks learning less. Although classification performance on transitioning motions was relatively lower than that for primary motions, they are still well recognizable as separate motions. Moreover, the implemented LSTM networks are useful enough because the primary motions are the main elements to consider for safety and productivity. Here, the transitioning motions are boundary motions and have an important role in distinguishing one primary motion from another so that the classification accuracy on the primary motions can be improved.

Among the three LSTM networks, the second and third networks showed a slightly lower classification performance than the first network. This is because the portion of transitioning motions in the second and third datasets is higher than in the first dataset. During the asphalt road construction work, the workers frequently changed from bending to standing and vice versa. This causes not only an increase in the number of the transitioning motions but also an increase of noise in the data. Generally, datasets for motion recognition are discrete datasets. In other words, motion data are collected from the subjects performing a particular motion. However, the datasets collected from the actual construction workers are continuous, so changes in motion may lead to ambiguity in separating motions distinctively.

The networks successfully recognized motions from the raw data. In the existing approaches, statistical features are extracted from the raw data to generate more representative features. However, feature extraction is not required when using LSTM networks, because the networks inherently learn the features during the training process. Due to this ability to learn features directly from raw data, end-to-end learning can be achieved in implementing the

LSTM networks.

The proposed study has some limitations. First, hand motions, such as swinging and holding a tool or material, cannot be recognized using the developed networks. Because the two IMUs are located on the back of the neck and lower back, they target motions of the torso. Further studies will focus on developing a model that recognizes motions of both hands and torso. Second, some of the motions commonly observed from the workers, e.g., squatting, are not investigated. Although three case studies of highway construction projects were conducted in this study, none of them had tasks that required such motions. Finally, the number of workers who participated in the study was not large enough to reflect personal motion bias. Every person has unique motion patterns, although variances can be minor. This can cause a slight decrease in classification performance when a pre-trained motion recognition model is implemented in a new site. However, this limitation is expected to be resolved once the datasets are collected from more workers with different types of tasks in a future study.

Conclusion

This paper proposes two-stacked LSTM networks for recognizing construction workers' motions. To practically validate the networks, three case studies were conducted in the actual construction sites. The three LSTM networks showed accuracies of 97.6%, 95.93%, and 97.36% on the testing datasets, respectively. Through the case studies, the technical and practical feasibility were investigated, and it was concluded that the networks properly recognized the motions of actual construction workers. With the developed motion recognition models, it is expected that individual workers' behavior and working conditions regarding safety and productivity can be automatically monitored and managed without excessive manual

observation.

The main contribution of this study is three folds. First, this study validated the technical and practical feasibility of the LSTM networks through real-world experiments with actual workers. The case studies conducted at three different construction sites showed that the networks are capable of learning characteristics of the motions of the workers in different tasks. Second, through the case studies, this study proved that the motion recognition method utilizing the LSTM networks can be implemented with the minimized constraints, which are 1) system instruction or guidelines are not required for the workers and 2) only two motion sensors are enough to achieve the expected accuracy for highway construction and maintenance applications. These advantages allow the system to be practically deployed in any construction project. Lastly, this study identified what kind of motions are involved in the application and how they are distributed between their jobs. With the system, safety and productivity are expected to be effectively managed by automatically recognizing workers' motions and working conditions.

Future studies will focus on integrating the networks with the location tracking methods developed by the authors (Cho et al. 2010; Fang et al. 2016; Park and Cho 2017). In addition to motions, locations can be important information to identify the safety and productivity of workers, because the range of possible working conditions can be narrowed by considering locations. Moreover, the networks can be further improved once the dataset is collected from more workers with different types of motions.

Data Availability Statement

Some or all data, models, or codes that support the findings of this study are available

from the corresponding author upon reasonable request.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No. (#1919068) and the Georgia Department of Transportation (GDOT) (RP18-17). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation and the Georgia Department of Transportation.

Reference

Akhavian, R., and Behzadan, A. H. (2016a). "Productivity Analysis of Construction Worker Activities Using Smartphone Sensors." *Proceedings of 16th International Conference on Computing in Civil and Building Engineering*, 1067–1074.

Akhavian, R., and Behzadan, A. H. (2016b). "Smartphone-based construction workers' activity recognition and classification." *Automation in Construction*, Elsevier, 71, 198–209.

Alwasel, A., Sabet, A., Nahangi, M., Haas, C. T., and Abdel-Rahman, E. (2017). "Identifying poses of safe and productive masons using machine learning." *Automation in Construction*, Elsevier B.V., 84, 345–355.

Antwi-Afari, M. F., Li, H., Seo, J., and Wong, A. Y. L. (2018). "Automated detection and classification of construction workers' loss of balance events using wearable insole pressure sensors." *Automation in Construction*, 96, 189–199.

Boyd, K., Eng, K. H., and Page, C. D. (2013). "Area under the Precision-Recall Curve: Point Estimates and Confidence Intervals." Springer, Berlin, Heidelberg, 451–466.

Chen, J., Qiu, J., and Ahn, C. (2017). "Construction worker's awkward posture recognition through supervised motion tensor decomposition." *Automation in Construction*, Elsevier, 77, 67–81.

Cho, Y. K., Youn, J. H., and Martinez, D. (2010). "Error modeling for an untethered ultra-wideband system for construction indoor asset tracking." *Automation in Construction*, 19(1), 43–54.

CPWR. (2018). *The Construction Chart Book: the U.S. construction industry and its workers*.

Fang, Y., Cho, Y. K., Zhang, S., and Perez, E. (2016). "Case Study of BIM and Cloud–Enabled Real-Time RFID Indoor Localization for Construction Management Applications."

Journal of Construction Engineering and Management, 142(7), 05016003.

Ghate, P. R., More, A. B., and Minde, P. R. (2016). *Importance of Measurement of Labour Productivity in Construction*. *IJRET: International Journal of Research in Engineering and Technology*.

Jebelli, H., Ahn, C. R., and Stentz, T. L. (2015). "Comprehensive Fall-Risk Assessment of Construction Workers Using Inertial Measurement Units: Validation of the Gait-Stability Metric to Assess the Fall Risk of Iron Workers." *Journal of Computing in Civil Engineering*, 30(3).

Kim, K., Chen, J., and Cho, Y. K. (2019). "Evaluation of Machine Learning Algorithms for Worker's Motion Recognition Using Motion Sensors." *Proceedings of the ASCE 2019 International Conference on Computing in Civil Engineering (i3CE)*, Atlanta, GA, USA.

Kim, K., and Cho, Y. K. (2020). "Effective inertial sensor quantity and locations on a body for deep learning-based worker's motion recognition." *Automation in Construction*, Elsevier B.V., 113, 103126.

Kingma, D. P., and Lei Ba, J. (2015). "ADAM: A Method for Stochastic Optimization." *the 3rd International Conference for Learning Representations*, San Diego.

McKinsey&Company. (2015). "The construction productivity imperative." *McKinsey&Company*, <https://www.mckinsey.com/industries/capital-projects-and-infrastructure/our-insights/the-construction-productivity-imperative> (Jan. 22, 2018).

McKinsey Global Institute. (2017). *Reinventing construction: a route to higher productivity*.

Nath, N. D., and Behzadan, A. H. (2017). "Construction Productivity and Ergonomic Assessment Using Mobile Sensors and Machine Learning." *Computing in Civil Engineering 2017*, American Society of Civil Engineers, Reston, VA, 434–441.

Ordóñez, F., and Roggen, D. (2016). “Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition.” *Sensors*, Multidisciplinary Digital Publishing Institute, 16(1), 115.

Park, J., and Cho, Y. K. (2017). “Development and Evaluation of a Probabilistic Local Search Algorithm for Complex Dynamic Indoor Construction Sites.” *Journal of Computing in Civil Engineering*, 31(4), 04017015.

Pei, L., Guinness, R., and Kaistinen, J. (2015). “Cognitive Phone for Sensing Human Behavior.” *Encyclopedia of Mobile Phone Behavior*, Z. Yan, ed., IGI Global, Hershey, PA, USA, 1138–1150.

Ryu, J., Seo, J., Jebelli, H., and Lee, S. (2019). “Automated Action Recognition Using an Accelerometer-Embedded Wristband-Type Activity Tracker.” *Journal of Construction Engineering and Management*, 145(1), 1–14.

Ryu, J., Seo, J., Liu, M., Lee, S., and Haas, C. T. (2016). “Action Recognition Using a Wristband-Type Activity Tracker: Case Study of Masonry Work.” *Construction Research Congress 2016*, 790–799.

Su, X., Tong, H., and Ji, P. (2014). “Activity recognition with smartphone sensors.” *Tsinghua Science and Technology*, 19(3), 235–249.

The National Institute for Occupational Safety and Health (NIOSH). (2019). “Ergonomics and Musculoskeletal Disorders.” *Workplace Safety&Health Topics*, <https://www.cdc.gov/niosh/topics/ergonomics/default.html> (Jul. 18, 2019).

Valero, E., Sivanathan, A., Bosché, F., and Abdel-Wahab, M. (2017). “Analysis of construction trade worker body motions using a wearable and wireless motion sensor network.” *Automation in Construction*, Elsevier, 83, 48–55.

Yan, X., Li, H., Li, A. R., and Zhang, H. (2017). "Wearable IMU-based real-time motion warning system for construction workers' musculoskeletal disorders prevention." *Automation in Construction*, Elsevier B.V., 74, 2–11.

Yang, K., and Ahn, C. R. (2019). "Inferring workplace safety hazards from the spatial patterns of workers' wearable data." *Advanced Engineering Informatics*, 41, 100924.

Yang, K., Ahn, C. R., and Kim, H. (2019). "Validating ambulatory gait assessment technique for hazard sensing in construction environments." *Automation in Construction*, 98, 302–309.

Yang, K., Ahn, C. R., Vuran, M. C., and Aria, S. S. (2016). "Semi-supervised near-miss fall detection for ironworkers with a wearable inertial measurement unit." *Automation in Construction*, 68, 194–202.

Zhao, J., and Obonyo, E. (2019). "Convolutional Long Short-Term Memory Model for Recognizing Postures from Wearable Sensor." *CEUR Workshop Proceedings*.

Zhao, Y., Yang, R., Chevalier, G., and Gong, M. (2018). "Deep Residual Bidir-LSTM for Human Activity Recognition Using Wearable Sensors." *Mathematical Problems in Engineering*, 2018, 13.