www.itcon.org - Journal of Information Technology in Construction - ISSN 1874-4753

COUPLING HUMAN ACTIVITY RECOGNITION AND WEARABLE SENSORS FOR DATA-DRIVEN CONSTRUCTION SIMULATION

SUBMITTED: December 2016
REVISION 1: June 2017
REVISION 2: November 2017
REVISION 3: January 2018
PUBLISHED: January 2018 at https://www.itcon.org/2018/1
EDITOR: Amor R.

Reza Akhavian, Assistant Professor,
Construction Management Program, School of Engineering, California State University East Bay;
Email: reza.akhavian@csueastbay.edu

Amir H. Behzadan, Clark Construction Endowed Associate Professor,
Department of Construction Science, Texas A&M University, College Station, TX 77843, USA;
e-mail: abehzadan@tamu.edu

SUMMARY: *Construction discrete event simulation (DES) models consist of a complex system of interrelated activities. An essential step in DES design is the process of input modeling which entails the estimation of the attributes (e.g. duration, precedence logic) of simulated activities. The quality of the simulation output is directly proportional to the quality of the input modeling. Traditional simulation models are commonly built upon engineering assumptions and subject matter expert opinions of simulation parameters. In this paper, a machine learning-based framework is designed and implemented to extract durations of activities performed by construction workers from wearable sensors. This framework uses accelerometer and gyroscope sensors embedded in smartphones that are worn by field workers. Data analysis and processing is applied to the collected data to train machine learning algorithms capable of detecting and classifying construction workers' activities. Once the activities are identified, their durations are calculated using the time stamps of the collected data. Results indicate that smartphones can be used as cost-effective, ubiquitous, and computationally powerful means of enabling data-driven DES models with enhanced reliability over traditional simulation models.*

KEYWORDS: *construction, data-driven simulation, sensors, wearable, smartphone, machine learning.*

REFERENCE: *Reza Akhavian, Amir H. Behzadan (2018). Coupling human activity recognition and wearable sensors for data-driven construction simulation. Journal of Information Technology in Construction (ITcon), Vol. 23, pg. 1-15, http://www.itcon.org/2018/1*

1. INTRODUCTION

Simulation, in the broadest sense, attempts to replicate the components and attributes of a real system in order to study its behavior under various conditions. Discrete event simulation (DES) is commonly used to achieve the same goal in planning, coordination, and management of construction projects (Halpin and Riggs, 1992). In a DES model, a number of resources with different quantitative and qualitative attributes are modeled to enable and feed a series of interconnected activities required to carry out a project. Accurate assignment of the attributes (e.g. type, amount) of such resources and activities (e.g. duration) result in a more realistic representation of the real system in the DES model (Banks, 2005). However, this introduces challenges in construction simulation due to the dynamic and transient nature of activities that makes it difficult to capture the true state of resources and activities. According to the literature, one way to address this issue is to collect factual data and establish the model based upon the real-world data (Song and Eldin, 2012, Akhavian and Behzadan, 2013).

Context-aware data collection from construction resources has been the subject of several previous studies (Akinci et al., 2006, Taneja et al., 2011, Shahandashti et al., 2011). In addition to the early attempts to track project resources using localization techniques that deployed sensors such as the Global Positioning System (GPS), vision-based detection and sensing have been also evaluated in construction settings (Jog et al., 2011, Gong and Caldas, 2010). In the past few years, however, unmatched capabilities of smartphones as pervasive, low-cost, and computationally powerful gadgets with robust communication and sensing capabilities have made them an unequivocal choice for object tracking on construction jobsites. Embedded smartphone sensors can help gather valuable information about the surrounding context. Among others, such sensors can collect data that enable identification of activities performed by the person who carries the phone (Brezmes et al., 2009). With roots in computer science, activity recognition is an emerging area of research that provides unprecedented opportunities for various application areas (Ravi et al., 2005, Casale et al., 2011, Favela et al., 2007, Ahn et al., 2013, Mathur et al., 2015).

In this research, smartphones are used to collect data from activities performed by construction workers in the field. In particular, accelerometer and gyroscope sensors are used to detect activities' start and finish points in time which in turn help calculate their durations. To assess the effectiveness of the designed methodology, a relatively complex operation consisting of multiple crews and several processes is simulated using operational knowledge extracted from smartphone data. Activities in this operation are carried out by workers who interactively contribute to individual processes. To have a true benchmark for obtaining real-world process-level data as well as verifying and validating the simulation results, the operation is also replicated in an experimental setting. Experimental workers were instructed to wear an armband that carried a smartphone. Accelerometer and gyroscope data were collected in real time using these smartphones.

Following the completion of the operation and data collection, the entire operation is modeled in a DES environment. The choice of DES for this purpose is due to the fact that most construction and infrastructure projects consist of intertwined yet discrete activities or sub-systems. In particular, two separate DES models are created; in model 1 (data-driven model) activity durations are extracted from sensor data using an activity recognition framework described in Section 2, whereas in model 2 (static model) activity durations are estimated based on certain heuristics (e.g. instructions given to workers during the experiment, physical dimensions of the workspace, approximated movement speeds). Having two different simulation models (where one uses extracted knowledge from sensor data, and the other follows the traditional process of input estimation) helps systematically compare the effectiveness of the developed methodology. In stochastic simulation models, probability distributions are fit to the activity duration values and used to describe that activity in the model. Therefore, while one set of duration data points is estimated and the other set is extracted, probability distributions are fit to each and such probability distributions are used in the simulation model.

2. CONTRIBUTIONS TO THE BODY OF KNOWLEDGE

Labor work and human interactions introduce a higher level of complexities to construction processes. For instance, an earthmoving operation entailing excavators and dump trucks can be carried out in a structured setting. However, in a labor-intensive scenario, interactions among human workers as well as their relative liberty in accomplishing tasks in non-repetitive and sometimes unscripted manners are not as predictable. Human body benefits from more degrees of freedom in performing physical tasks than a piece of heavy construction equipment.

Moreover, tracking functionality, productivity, and safety aspects of equipment is more trivial through the use of various monitoring tools and onboard instruments, which may or may not be readily applicable to human motion tracking. The authors have previously focused their efforts on detecting activities performed by heavy machinery using equipment-mounted smartphones (Akhavian and Behzadan, 2015). Having achieved successful results in their past work, the new focus area of authors' work is designing and testing robust and easy-to-use techniques for detecting field activities carried out by human crews through the application of mobile sensors (e.g. smartphones). This enables the creation of a comprehensive and practical methodology that can be replicated in a variety of construction projects where human workers, equipment, or a combination of both are deployed to carry out project tasks. With this important goal in mind, the findings of this work are ultimately sought to be integrated into a data-driven simulation framework that allows the generation of responsive and constantly adapting simulation models linked to the process-level knowledge extracted from mobile sensor data. The creation and maintenance of more realistic and reliable simulation models have in fact been identified as a grand challenge by the American Society of Civil Engineers (ASCE) Visualization, Information Modeling, and Simulation (VIMS) Committee (Leite et al., 2016). Therefore, this research directly supports the prospect of the widespread use of simulation-based decision-making in the construction industry. In summary, the key scientific and practical contributions of this paper to the body of knowledge are to lay the foundation, assess the significance, and report the results of data-driven simulation model development through activity recognition in labor-intensive construction operations. It should be noted that the computational contributions of this study lay in the design and implementation of a data-driven simulation framework using smartphone sensors. The prerequisite step to this effort is the development of a robust activity recognition framework the details of which have been described at length in another publication by the authors (Akhavian and Behzadan, 2016). This paper mainly aims at (1) evaluating the feasibility of discovering process-level knowledge from smartphone sensor data that is interpretable in a DES model, and (2) confirming that the simulation output generated from a data-driven (i.e. dynamic) DES model more closely mimics the real-world process compared to the simulation output generated by existing (i.e. static) DES modeling techniques.

3. ACTIVITY RECOGNITION FRAMEWORK

Smartphone accelerometer and gyroscope are very sensitive to the movements of the device. The accelerometer is a sensor that measures acceleration and gyroscope measures angular velocity in the 3D space (i.e. X, Y, and Z). While the smartphone is mounted on (i.e. worn by) the user, his or her body motions are captured by the embedded sensors within the phone's coordinate frame. Since each activity requires certain body parts to move with a specific pattern in various directions in space, accelerometer and gyroscope sensors produce identifiable signals for any given physical activity. This is the underlying principle of activity recognition using smartphones.

The use of accelerometer only, accelerometer plus gyroscope, and to a lesser extent, accelerometer and gyroscope in a smartphone have been previously evaluated in computer science for medical, public health, and sports applications (Favela et al., 2007, Bao and Intille, 2004). More recently, within the construction domain, researchers have evaluated applications that include activity analysis of workers (Joshua and Varghese, 2010), safety of ironworkers (Yang et al., 2016), and equipment activity recognition (Akhavian and Behzadan, 2015). The activity recognition framework in this study encompasses a thorough and novel design that enables activity duration extraction for simulation modeling. The following briefly describes the designed framework components and corresponding procedures. Data is collected using commercially available smartphone apps on the Apple Store and Google Play that collect and log timestamped data. The app that was used to collect data is called Sensor Kinetics Pro by Innoventions® which is available to both iOS and Android operating systems. There are a number of other apps available on the Apple Store and Google Play but this app enables collecting data from multiple sensors (i.e. accelerometer and gyroscope) at the same time. Moreover, the results are logged into a spreadsheet and can be emailed to a specified address immediately. Also, the app allows calibrating the sensors on a fixed surface so it improves the reliability. The app has shown more robustness in terms of drift compared to other apps because of its sensor fusion capabilities. The logger app exports data in comma-separated values (.csv) format in a spreadsheet, which is then used as an input file for analysis in Matlab. All the steps taken for activity recognition including feature extraction and supervised learning were conducted in Matlab software. More detailed discussion about authors' work on activity recognition and evaluation of alternatives can be found in (Akhavian and Behzadan, 2015).

The developed activity recognition framework in this research employs machine learning algorithms that can be designed using supervised or unsupervised training methods and used for regression or classification analysis. In this study, supervised classification algorithms are used to detect activities of construction workers. Supervised machine learning classifiers map certain patterns in independent variables (i.e. features) to the known dependent variables (i.e. classes or labels) for the purpose of model training. Once the model is trained, it is expected to be able to predict the classes by observing data points that were not used in the training process. Features are statistically derived from collected sensor data and used to train algorithms that map signal patterns to activity classes.

Collected data are segmented into windows each containing a certain number of data points. In this research, 100 data points are collected per second (i.e. data collection frequency of 100 Hz) and every 128 data points (or collected data in 1.28 seconds) are segmented in one window. The choice of 128 data points was made for two reasons. First, because of using fast-Fourier transform (FFT) technique to derive frequency-domain features, the number of data points must be a power of 2. Second, considering the nature of activities performed, 1.28 seconds provides a suitable divisor when breaking down activity durations. However, it is worth mentioning that depending on the nature of activities, different data points per window and different frequencies can be used. Also, the frequency of 100 Hz was used for data collection to ensure capturing all body movements. This may be considered as a high frequency for the bandwidth of the movement data of interest. However, considering the computing and storage power of today's computers, using 100 Hz frequency does not present any problem and only guarantees an inclusive (high resolution) data collection. Commonly used statistical features in sensor-based activity recognition such as mean, minimum, maximum (i.e. time-domain features), and signal entropy and energy (i.e. frequency domain) are then formed in each window. A detailed description of the employed features is provided in Table 1.

The extracted features shown in Table 1 are used as the input of the machine learning algorithms. In particular, five supervised classifiers are chosen, trained, and then used to predict construction workers activities. The selected classifiers include feed-forward backpropagation artificial neural network (ANN), support vector machines (SVM), K-nearest neighbors (KNN), logistic regression, and decision trees.

TABLE 1. Features used in the activity recognition from accelerometer and gyroscope data in x, y, and z axes (p_i = sensor data points).

Feature Type	**Feature Name**	**Mathematical Expression**
Time-Domain	Mean (in each axis)	$\frac{\sum_{i=1}^{N} p_i}{N}$
	Minimum	$\min\lvert p_i \rvert$
	Maximum	$\max\lvert p_i \rvert$
	Variance	$\frac{\sum_{i=1}^{N}(p_i-\mu)^2}{N}$ *where μ is the mean*
	Root mean square (RMS)	$\sqrt{\frac{\sum_{i=1}^{N} {p_i}^2}{128}}$
	Interquartile range (IQR)	$Q_3 - Q_1$ *where Q_i is the ith quartile*
	Correlation (between each two axes pair)	$\frac{\sum_{i=1}^{N}(p_i-\bar{p})(m_i-\bar{m})}{\sqrt{\sum_{i=1}^{N}(p_i-\bar{p})^2}\sqrt{\sum_{i=1}^{N}(m_i-\bar{m})^2}}$
Frequency-Domain	Spectral Energy	$\sum_{n=1}^{N}\lvert p(n)\rvert^2$
	Spectral Entropy	$-\sum_{l=1}^{N} P(l)\log\big(P(l)\big)$ *where P(l) is power spectral density*

After a detailed evaluation of the five classifiers, ANN showed the best performance in terms of classification accuracy through a 10-fold cross validation, followed by the KNN. Therefore, both classifiers were combined through an ensemble learning methodology called the bootstrap aggregating or Bagging. In Bagging, T training data subsets each containing m training examples are selected randomly with replacement from the original training set of m examples. The classification result of the ensemble is determined through plurality voting (Lin et al., 2003). For the ensemble learning method used in the experiments for this research, the number of training dataset is $T = 10$. Different activities are recognized and classified within the three process categories. Table 2 summarizes the process categories and activities used for recognition and classification.

TABLE 2. List of the processes involved in the operation and activities within each process.

Category	Process	Activity/State
1	Cutting Lumber	1. Sawing
		2. Idle
2	Installation	1. Hammering
		2. Turning the Wrench
		3. Idle
3	Transportation	1. Loading
		2. Hauling
		3. Unloading
		4. Returning
		5. Idle

In each process category, activities shown in Table 2 (in addition to an Idle state of the workers) are evaluated for classification. For example, within the Installation process category, the classifier accuracy in detecting Hammering, Turning the Wrench, and Idle activity/state is evaluated. Results are then evaluated by considering the number of correctly identified instances of activities over total instances of activities in that process category. This assessment is done for all five classifiers and then for the ensemble model. Accuracy, precision, and recall have been used as three measures for assessing the classification results. Equation 1-3 show how each value is calculated.

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad \text{(Equation 1)}$$

$$\text{Precision} = TP / (TP + FP) \quad \text{(Equation 2)}$$

$$\text{Recall} = TP / (TP + FN) \quad \text{(Equation 3)}$$

In Equations 1, 2, and 3, for any given *Activity i*: TP is the "true positive" that refers to the number of instances of *Activity i* that were correctly classified as *Activity i.* TN is the "true negative" that refers to the number of instances of *other Activities* that were correctly classified as *other Activities.* FP is the "false positive" that refers to the number of instances of *other Activities* that were incorrectly classified as *Activity i.* FN is the "false negative" that refers to the number of instances of *Activity i* that were incorrectly classified as *other Activities.*

Table 3 shows the accuracy results of all three process categories for each classifier and the ensemble model for all three categories. In addition, precision and recall values are calculated for the ensemble model as the best classifier. It is worth mentioning that precision and recall are usually used in binary classification problems, but in this case, are calculated in a one-versus-all manner for more detailed assessment of the classification performance. Results are presented in Table 4 for each activity/state according to Table 2. As it is shown in Table 4, there is a decline in the performance of the classifier in category 3. The lower accuracy is attributed to the complexity as well as the increase in the number of activities in the third category. Moreover, the activities of the third category constitute similar movements in the body parts (e.g. Hauling vs. Returning and Loading vs. Unloading) and that affects the ability of the algorithm to distinguish them in the classification process.

TABLE 3. Classification accuracy results for different classifiers and the ensemble model (%).

Category	**Classification Method**					
	ANN	**KNN**	**Logistic Regression**	**Decision Tree**	**SVM**	**Ensemble Model**
1	96.27	95.58	96.54	95.58	96.64	99.28
2	87.78	78.57	82.23	78.57	82.18	90.09
3	88.17	85.62	85.84	85.62	78.34	92.97
Average	90.74	90.54	88.2	86.59	85.72	94.11

TABLE 4. Precision and recall for the Ensemble Model for each activity in different categories (%).

Category	**1**		**2**			**3**				
Activity	1	2	1	2	3	1	2	3	4	5
Precision	97.87	98.78	87.89	86.64	84.94	44.93	59.47	56.30	53.26	92.70
Recall	97.18	99.08	97.79	94.61	79.75	69.75	86.27	91.08	85.68	97.07

It should be noted that the activity recognition and duration extraction were carried out in sequential automated steps, but the integration of the steps from recognition of the activities to inputting the durations into the DES model was done manually.

4. DESIGN OF THE VALIDATION EXPERIMENT

In this section, the experiment setup and description of the activities involved are discussed. Next, the modeling platform and simulation design of the operation are described which is followed by the results of activity duration extraction. Finally, a comparative analysis of static versus data-driven simulation models is carried out and conclusions are made.

4.1 Experiment Setup

The operation to be simulated consists of preparing, transporting, and installing wood sections. An outdoor environment that resembled a small construction jobsite was chosen to conduct the experiment. FIG. 1 shows a schematic illustration of the operation. As shown in FIG. 1, the cyclic operation starts with a worker, labeled as *W1*, who saws lumber inside a *wood workshop* and prepares wood sections of proper sizes and shapes. These sections are then transported to the *installation area* by two other workers, labeled as *W2* and *W3* who are tasked with loading the sections, hauling them to the installation area, and dumping them where an installer worker, labeled as *W4*, is waiting to receive the sections and install them in their positions. FIG. 2 is a real photo of the experiment.

Each process involves one or more activities assigned to different workers. In the *wood workshop,* the process of cutting the lumber pieces consists of only one activity, *sawing*, carried out by worker *W1*. The transportation process involves four activities namely putting sections into the wheelbarrow or *loading,* pushing a loaded wheelbarrow or *hauling*, dumping the sections in the installation area or *unloading*, and returning the empty wheelbarrow or *returning*. Workers *W2* and *W3* are responsible for the transportation process. Finally, worker *W4* is tasked with the installation process which involves the activities *hammering* and *turning the wrench*. The experiment was conducted for 30 cycles, meaning that the number of observations for each activity was 30 times for Cutting Lumber, and Transportation categories and 60 times for the activities in the Installation category. The latter contained Hammering and Turning the Wrench activities on a half piece of a lumber section that was cut into half.

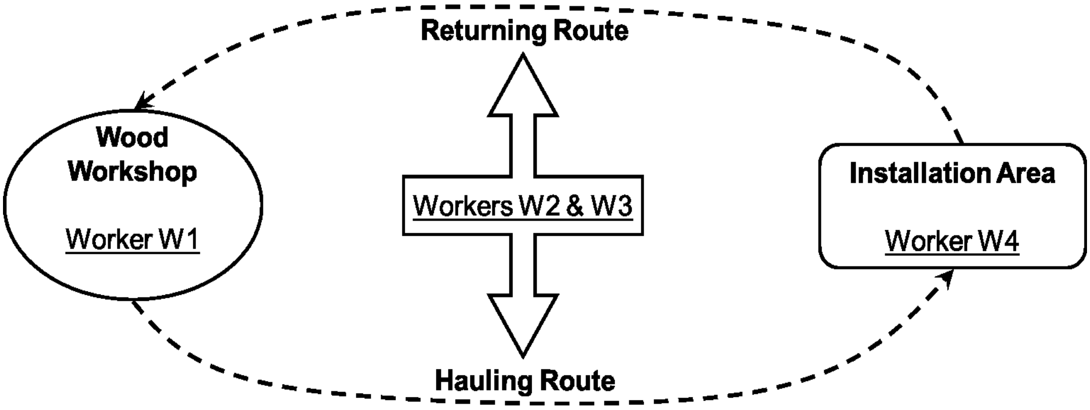


FIG. 1: Schematic illustration of the operation experiment setup.

FIG. 2: A photo of the operation showing four workers performing the experiment.

The *loading* and *unloading* activities follow underlying operational rules that are enforced in the experiment and later in the simulation models. The first operational rule is applied to the *loading* activity; in particular *loading* will not be executed until there are at least two wood sections available for transportation. Therefore, when there are less than two sections prepared by worker *W1,* and either or both workers *W2* and *W3* are available, they will have to wait in a queue until at least two sections are ready for loading. With the same token, if either or both workers *W2* and *W3* are available, one section should wait until there is at least one more section prepared by worker *W1* so that both sections can be loaded. The second operational rule is for *unloading*. It is assumed that the space available for unloaded sections is enough only for two sections and *unloading* activity should be executed in only one instance, meaning that if there is any section waiting to be processed by worker *W4*, the available workers *W2* or *W3* should wait until there is no section awaiting the installation process. The last operational rule that is applied to both *loading* and *unloading* activities is that only one instance of each activity can be performed at any given time, meaning that simultaneous execution of either *loading* or *unloading* activity is not allowed. It should be noted that these rules are considered as assumptions to control the experiment environment and may or may not be necessarily held true in real-world operations.

4.2 Simulation Model of the Operation

The operation described above was meticulously modeled in STROBOSCOPE (STate and ResOurce Based Simulation of COnstruction ProcEsses), a DES scripting environment based on Activity Cycle Diagrams (ACDs) that is designed for the simulation of processes common to construction engineering (Martinez, 1996). Simulation models created in STROBOSCOPE are based on a network of interconnected modeling elements described in a script containing programming statements that give the elements unique behavior and control the simulation (Martinez and Ioannou, 1994). This network of the interconnected elements (a.k.a. the ACD) is designed to be similar in appearance and function to CYCLONE simulation platform, which was the first system developed specifically for construction operations (Halpin, 1977). The ACD of the operation described in Section 4.1 is shown in FIG. 3.

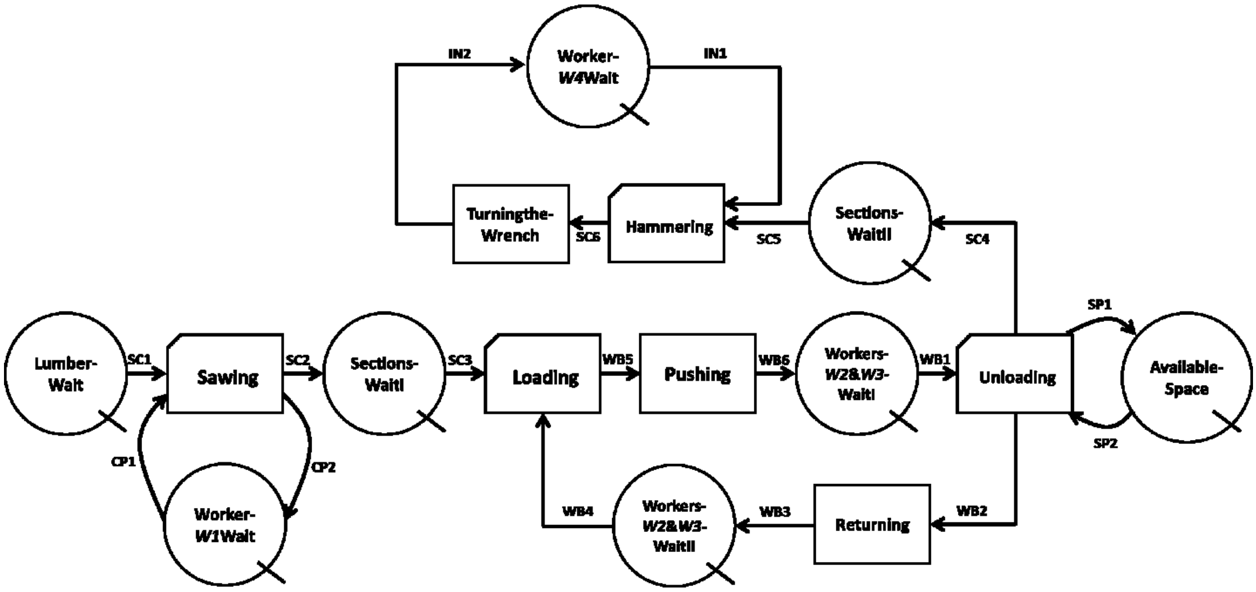


FIG. 3: The ACD of the operation for modeling in Stroboscope.

In FIG. 3, resources move from each node to the succeeding node in the direction shown by the connection link. A circle with a slash in the bottom right corner is a *Queue* that serves as the storage location for the resources. A rectangle with a cut-off in the top-left corner is called a *Combi* and a regular rectangle is called a *Normal*. These two nodes represent two different types of activities and hold the resources for the amount of time determined by activity durations. In particular, a *Combi* is always preceded by a *Queue* while a *Normal* activity cannot be preceded by a *Queue*. In FIG. 3, `LumbersWait` holds lumber pieces before they are taken by worker *W1* for activity `Sawing`. The `WorkerW1Wait` Queue populated with 1 entity (i.e. 1 worker) ensures that only one instance of the `Sawing` activity is carried out in any point of time. Upon being sawed, sections wait in `SectionsWaitI` Queue to be loaded for transportation. This Queue satisfies the first operational rule (described earlier). The `WorkersW2&W3WaitII` Queue is where Workers *W2* and *W3* are drawn to load only two sections, if available. Similar to `SectionsWaitI,` this Queue also contributes to satisfying the first operational rule. When enough sections and transportation workers are available, the `Loading` Combi is activated, lasts for its assigned duration, and then releases the captured resource (i.e. worker) to the `Hauling` Normal. Again, this activity will hold the resource for the amount of time determined by its corresponding duration. Next, according to the second operational rule (described in Section 4.1), Workers *W2* and/or *W3* wait in the `WorkersW2&W3WaitI` Queue before the space is available for activation of the `Unloading` Combi. Finally, the `SectionsWaitII` Queue is where at most two sections are being held before they can proceed to the `Hammering` Combi. It must be noted that the `Hammering` Combi will not be activated if either of the `SectionsWaitII` or `WorkerW4Wait` Queues does not have available resources. Such a situation happens for example if worker *W4* is captured by the `TurningtheWrench` Normal.

To ensure that the developed ACD and the operations in the real-world follow a similar precedence logic and potential variations are within an acceptable range, a point-by-point assessment of the ACD and the real system was conducted at random timestamps. In particular, worker activities were videotaped and annotated. This log was then compared with the future event list (FEL) generated by the simulation model at random points in time.

While the ACD that is shown in FIG. 3 provides a high level representation of the simulated operation, more specific operational details are incorporated in the script of the model. This is where attributes of the queues and activities as well as the model parameters are assigned. Such attributes define how model parameters behave. For example, the followings show how some of the network elements inducing activities (Normal and Combi), Queues, and links are defined in Stroboscope:

```
/* Definition of Network Elements
QUEUE SectionsWaitI     Sections;
COMBI Loading;
NORMAL      Hauling;
QUEUE WorkersW2&W3WaitI Workers;
COMBI Unloading;
LINK  CP1   WorkerW1Wait      Sawing;
LINK  CP2   Sawing            WorkerW1Wait;
LINK  SC1   LumberWait  Sawing;
LINK  SC2   Sawing            SectionsWait1;
```

Other key attributes are the durations of Combi and Normal activities. Activity durations are sampled from the specified probability distributions. In Section 4.3, the activity recognition framework developed in this research in order to extract realistic activity durations is described.

4.3 Duration Extraction through Activity Recognition

The activity recognition framework described in Section 3 is the basis for duration extraction. The classification was performed on the activity level within each process, meaning that the result of the classification in terms of accuracy in correctly predicting the activities within each process is reported. It must be noted that within each class, an extra activity is included as the *idling* state in which the worker is not contributing to any of the assigned activities within the process. Activity duration extraction particularly used the number of windows detected that correspond to each instance of an activity. Since those windows are timed (i.e. each contains 1.28 seconds worth of activity duration), they are used as the basis for duration extraction. However, the process involves some complexities that were resolved using heuristic algorithms described in this section.

As seen in Table 3, the classification accuracy for the first process involving the *sawing* activity is almost perfect and it is expected that it closely matches the observed activity durations. This will be assessed in a more statistically rigorous manner in Section 4.4. However, activities that comprise the other two processes, namely transportation and installation have not been classified as accurately. Therefore, the durations extracted from these activities are expected not to be as close to the observed durations as the first process. It should be noted that the similarity between the extracted durations and the observed values does not necessarily conform the same accuracy of their associated activity recognition. In other words, although it is expected that the durations of activities within the Cutting Lumber process are predicted with the highest accuracy of all, the accuracy of predicting activity durations for the Transportation and Installation processes may not follow the same results in terms of relative accuracies. This is due to the fact that extracting activity durations follows a heuristic algorithm according to which many of the misclassified instances are discarded. Nevertheless, the more accurate activity recognition indirectly translates to more accurate activity duration calculation. This signifies the role of activity recognition framework for the overarching goal of this study to extract true activity durations. The algorithm that is used to calculate activity durations first replaces instances of different classes that are appeared within a large number of detected instances of the same class. For example, few instances of class C_2 classified after many instances of class C_1 followed by other instances of class C_1 are all considered as class C_1. The exact numbers followed by this heuristic algorithm depends on the sampling frequency, window size, and a rough approximation of the activity durations. In this study, the sampling frequency is 100 Hz. Windows include 128 data points with 50% overlap that amounts to 0.64 seconds of data. Any two instances of an activity that normally take more than 20 seconds but are separated out to last for less than 12 seconds are merged. Such heuristics result in improved accuracy for activity duration extraction.

4.4 Simulation Input Modeling

In this section, the process of input modeling of the operation simulation is described. Simulation input modeling includes fitting probability distribution functions to the activity durations and is therefore, of the essence to the accuracy of the model. First, observed activity durations using the recorded videotape of the experiment are compared to those extracted through the activity recognition system. This step serves to guarantee that extracted activity durations are not statistically significantly different from those that actually took place in the real experiment. If there is a statistically significant difference between the two sets of duration values, then the validation of the model will not be accurate, meaning that it cannot be expected from the data-driven simulation model to output values close to the actual ones observed in the experiment. Extraction of the durations from the video recording was conducted manually by marking the start and finishing points of activities in the data collected to synch with the videotape.

In order to compare observed and extracted activity durations, the *student t-test* is used to evaluate the null hypothesis of no considerable difference between the expected and sample distributions. Table 5 shows the result of the t-test for activities within each process. As shown in this Table, the null hypothesis for none of the activities was rejected through comparison of the observed and extracted activity durations with 5% significance level. This confirms that the two sets of activity durations are not statistically significantly different.

The objective of creating simulation models of the operation experiment is to compare the results of the simulation created based on the extracted activity durations (data-driven model) to the one created according to the estimated activity durations (static model). To this end, estimated activity durations were defined by taking into account the [minimum, maximum], or three-point estimation [minimum, mode, maximum] durations for each activity which is a common practice in creating construction simulation models or project management schedules using project evaluation and review technique (PERT) (Halpin and Riggs, 1992). These two schemes are in essence equivalent to sampling from uniform and triangular distributions. Therefore, these two probability distributions were considered for activity durations inside the static model. The parameters of the two probability distributions, however, were estimated according to two heuristics; the instructions given to the workers performing the activities, and engineering assumptions of the variance for such durations considering the nature of each activity. For example, worker W1 was asked to saw each piece of lumber for about 25 to 30 seconds. Therefore, in the static simulation model, the probability distribution considered for this activity was a uniform distribution with a minimum of 22 and maximum of 33 to account for 3 seconds of variations from the extrema. As for the data-driven simulation model, probability distributions representing extracted activity durations were strictly chosen based on the rankings calculated by Kolmogorov–Smirnov and ChiSquare goodness-of-fit tests (Akhavian and Behzadan, 2014). Table 6 shows the probability distributions fitted to the extracted activity durations along with those estimated for each activity.

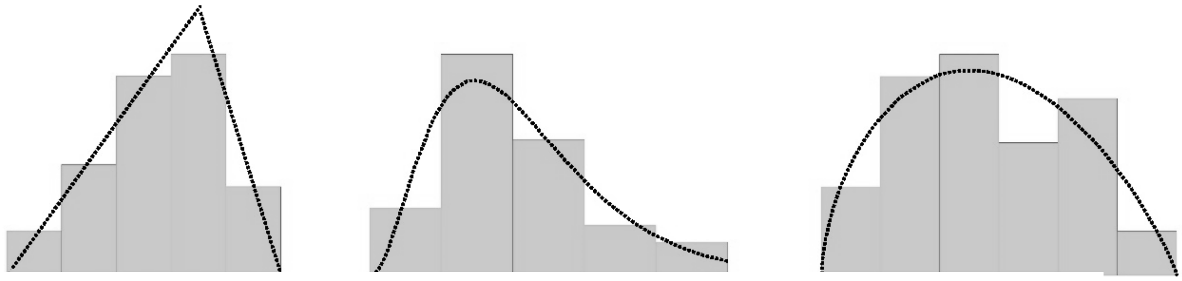
TABLE 5. Comparison of the observed and extracted activity durations using student t-test.

Process	Activity	Observed Duration (Seconds)		Extracted Duration (Seconds)		*p*-value	Null Hypothesis
		Mean	SD	Mean	SD		
Cutting Lumber	**Sawing**	27.95	6.50	27.97	6.57	0.78	Not rejected
Transportation	**Loading**	8.96	1.40	9.24	1.75	0.27	Not rejected
	Hauling	14.02	2.43	14.14	2.81	0.63	Not rejected
	Unloading	13.18	1.96	13.53	2.01	0.08	Not rejected
	Returning	11.33	2.14	11.39	2.29	0.78	Not rejected
Installation	**Hammering**	17.05	2.48	17.59	2.46	0.09	Not rejected
	Turning the Wrench	13.39	3.42	13.44	3.35	0.75	Not rejected

TABLE 6. Probability distributions used inside the two simulation models.

Activity	Probability Distributions Used for	
	Extracted Duration (Data-Driven Model)	**Estimated Durations (Static Model)**
Sawing	Triangular[12,31.6,40]	Uniform[22, 33]
Loading	Triangular[6,8.1,13]	Uniform[5, 7]
Hauling	9 + Gamma[1.63, 3.16]	Uniform[8, 12]
Unloading	9 + 8 × Beta[1.83, 1.41]	Uniform[7, 12]
Returning	6 + Gamma[1.26, 4.28]	Uniform[7, 10]
Hammering	Normal[17.8,2]	Triangular[13,15,17]
Turning the Wrench	7 + 14 × Beta[1.62, 1.78]	Triangular[13,15,17]

FIG. 4 shows the probability distributions fitted to the extracted durations for one activity per each category as a sample. The distribution fitting was performed using Arena® Input Analyzer software (RockwellAutomation, 2017). The activities shown are the ones in the extracted category to show how well the extracted data are represented using probability distributions. In addition, the activities shown are selected from different types of probability distributions as samples due to space limitations to show all 14. It is also worth mentioning that different numbers of bins have been used in the histograms where each bin contains a portion of the data points to represent the best-fit probability distributions found.



(a) (b) (c)

FIG. 4: Examples of the probability distributions fitted to the extracted durations for (a) Sawing, (b) Hauling, and (c) Turning the Wrench activities

4.5 Performance of the Data-Driven Model vs. Static Simulation Model

Using the two sets of probability distributions shown in Table 6, two identical simulation models are created based on the ACD introduced in FIG. 3. The main difference between the two simulation models is in the activity durations defined in the input script of each model. Each model was run for 50 replications by generating random numbers from the same seed. This means that during the 50 replications, the random values were drawn from an identical set of numbers. Also five measures were collected for comparison of the outputs of the simulations to the real-world observations. The measures include the average waiting times (in seconds) of the entities in the four Queues namely `SectionsWaitI`, `SectionsWaitII`, `WorkersW2&W3WaitI`, and `WorkersW2&W3WaitII`, as well as the total operation time (in minutes). FIG. 5 shows the comparison between the five measures. For each measure shown in FIG. 5, the first bar from the top refers to the value of that measure observed in the real-world operation. The second bar with a slightly lighter color corresponds to the mean of the average waiting times resulted from the data-driven simulation after 50 replications. The error bar (whiskers) refers to the standard deviation of these 50 replications. The third bar with the lightest color is the result of the static simulation created based on the estimated activity durations.

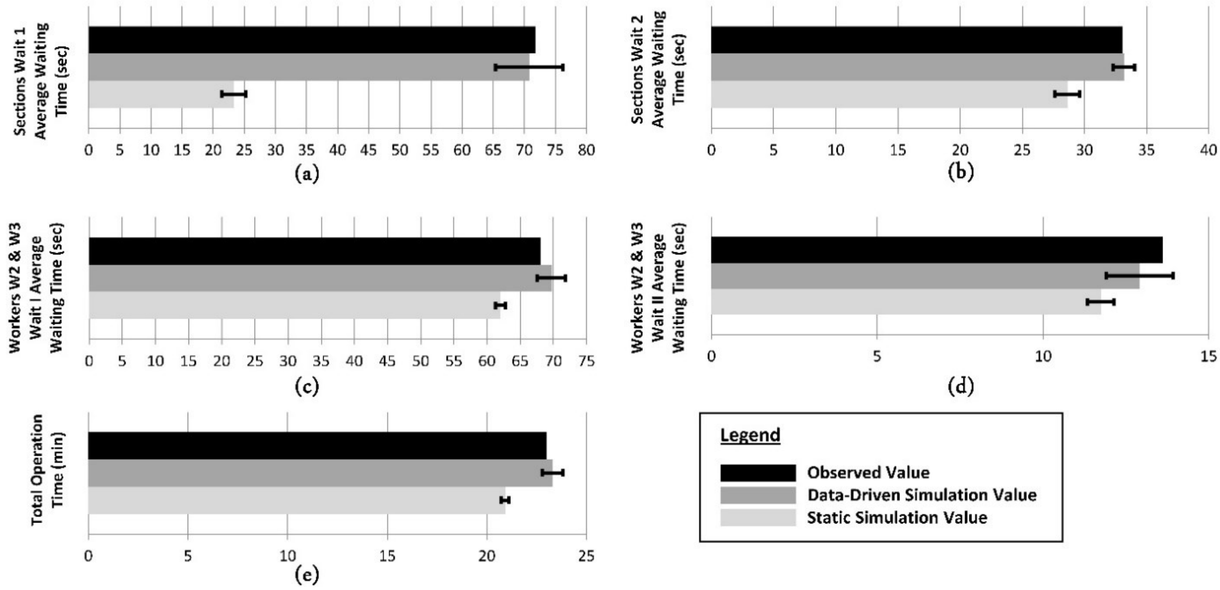


FIG. 5: Comparison of values obtained from the real-world experiment and the output of static and data-driven simulation models

4.6 Discussion of the Results

According to FIG. 5, the observed values for all five measures are within one standard deviation of the results obtained from the data-driven simulation model. However, all the output measures obtained from the static simulation with estimated values underestimate the actual values. In fact, this is what occurs very often in construction projects where simulation models created during the early planning and pre-construction stages significantly underestimate or overestimate the durations of the real-world processes (Halpin and Riggs, 1992). This happened while uniform and triangular probability distributions and not fixed values were used for estimating activity durations for the static model's input. It must be noted, however, that the underestimation observed in the output of the static simulation model in all five measures is particular to this specific experiment and may not be generalizable to other scenarios. More specifically, the measures obtained from the static model could have as well resulted in an overestimation. What is of utmost importance in interpreting the results is the noticeable difference between the outputs of the two simulation models and the fact that the result of the data-driven model is closer to real-world observations. It is worth mentioning that the true benchmark used to validate the results of the data-driven simulation model is the first bar (i.e. observed values). In other words, while being superior to the results of the static simulation model (i.e. third bar) is a must to demonstrate the success of the developed framework, it is imperative that the output of the data-driven simulation model be also close enough to observed values. The observed values were extracted by meticulously analyzing the videotape of the experiments and therefore, were at least as accurate as timer-based measurements (which themselves are subject to human errors inherent to any manual procedure).

A considerable discrepancy is observed in the result obtained from the static simulation and the observed value for the average waiting time in Queue `SectionsWait1`. This can be explained as follows; since the `WorkersW2&W3WaitII` average waiting times in bar chart (d) are very close (considering the scale of this chart), the availability of workers should not have influenced the difference. Therefore, it can be explained through the difference in durations considered for the `Sawing` activity. It turns out that the data-driven simulation with the probability distribution of Triangular [12, 31.6, 40] for `Sawing` activity, samples from a lower range of numbers starting from 12 seconds, while the minimum value for the uniform distribution of the observed values is 23 seconds. In reality, this results in a much faster `Sawing` activity which in turn provides more sections waiting in the `SectionsWait1` Queue. Other than `Sawing`, most of the other activities have estimated distributions resulting in the sampling of lower values that explains the low overall operation time obtained from the static simulation.

Regardless of the reasons for any discrepancy between the extracted and estimated activity durations, the very fact that any difference in activity durations can substantially change the simulation output statistics verifies the significance of having more realistic simulation models through data-driven input modeling.

5. LIMITATIONS

The assumptions and theoretical foundations of the research presented in this paper have been assessed in a medium-scale laboratory setting experiments. In doing so, the authors had to make a few simplifying (yet strongly supported by literature) experimental assumptions. The first assumption is that the field operation strictly follows the ACD shown in FIG. 3. It is conceivable that in an actual setting, deviation (even minor) may exist between the real-world operations and the ideal flow of activities shown in FIG. 3. Therefore, in a real-world setting, there might be other activities involved in completing a task, but the assumption here is that those activities do not follow a reparative pattern and are not considered as part of the value-adding movements to completion of the task. Another assumption that was made during the collection of sensor data is that human workers follow clear instructions when performing field activities. In the conducted experiments, while workers were told that small deviations from instruction in terms of, for example, activity durations are acceptable, they were still asked to adhere to instructions as closely as possible when performing their tasks. Finally, as discussed earlier, the assumption made to estimate activity durations for the static simulation model were mainly rooted in classic texts (Halpin and Riggs, 1992) and the heuristics according to which most simulation models are currently developed in practice. Therefore, it is understood that while the static simulation model (benchmark model) used in this work has been meticulously designed to mimic the real-world operations, there may be still some discrepancies between the generated results and the outcome of the actual field operations, that can potentially introduce bias into the analysis. Having said this, as part of the future directions of this work, the authors are planning to carry out larger-scale less-constrained experiments with more complexity to better study and ultimately eliminate this likely bias in data analysis. Moreover, given that a fully functional data-driven simulation framework is a powerful tool for enhancing productivity and safety, such applications are currently under review by the authors for further implementation. It is also worth mentioning that admittedly, data collection from human subjects comes with privacy concerns. In this study, standard data collection protocols prescribed by the institutional review board (IRB) were followed to collect anonymous data from construction workers. Collected data is solely used for the purpose of updating activity durations inside simulation models corresponding to the real-world operations. The aim is to make the simulation model adaptive to the changes on the ground, and as such, no private information is extracted from the collected data.

5.1 Threats to Validity of Research Design

It is important to evaluate the effectiveness of the applicability of the research and experiment design to avoid making false assumptions (Yu and Ohlund, 2010). In the presented research the Internal Validity might be affected by the Instrumentation through a change in smartphone device or app. In any case, to minimise such jeopardy of the internal validity, both dominant smartphone operating systems have been experimented. Also, the placement of the smartphone may affect the results and different sets of experimentation are required to locate the best body part that sufficiently captures all the activities. Selection of Subjects might also introduce some effects on the experimentation designs as the subjects in this research experiment were construction management graduate students and not real-world workers performing their daily tasks. To limit this effect, having real-world experience in a construction jobsite has been considered. In terms of the External Validity, the artificiality of the experimental setting may have introduced reactive effect. It is also worth mentioning that the experimental setting may lack activities that could occur in the real-world settings and if case of inclusion of such those activities, the resulted accuracies may be different. No other threats are identified in the research design.

6. SUMMARY AND CONCLUSIONS

In this paper, an operation involving multiple interactions between workers performing construction field activities was described and modeled in DES using process-level data collected from the crew in real time. Sensor data consisted of accelerometer and gyroscope data and were collected using smartphones affixed on workers' upper arms. Activities performed by workers were then recognized and classified using an activity recognition methodology designed by the authors and introduced in this paper. Following this activity recognition, activity durations were extracted and probability distributions were fit to these extracted durations. Moreover, extracted durations were compared to duration values observed in the real-world experiment to confirm their fidelity. Extracted activity durations were then fused into a data-driven DES model created based on the experiment design in order to compare the results against those of a similar but static simulation model with estimated activity duration values.

Analysis of the output obtained from the two simulation models with respect to five quantifiable measures (i.e. average waiting times of the entities in four Queues namely `SectionsWaitI`, `SectionsWaitII`, `WorkersW2&W3WaitI`, and `WorkersW2&W3WaitII`, as well as the total operation time) revealed that the data-driven simulation model created based on the knowledge (i.e. activity durations) extracted by the developed activity recognition framework outperforms the static simulation model created based on estimated activity durations. Considering that the current practice in creating construction simulation models is using historical (secondary) information and subjective assumptions in designing model attributes, obtaining results in close agreement with reality reaffirms the significance of substituting this traditional approach in creating simulation models with a more robust and reliable data-driven and knowledge-based methodology that was described in this paper. That being said, more research is needed to evaluate data-driven simulation frameworks in different construction contexts using more complicated scenarios and workflow conditions.

REFERENCES

Ahn, C. R., Lee, S. & Peña-Mora, F. (2013). The Application of Low-Cost Accelerometers for Measuring the Operational Efficiency of a Construction Equipment Fleet. *Journal of Computing in Civil Engineering*.

Akhavian, R. & Behzadan, A. H. (2013). Knowledge-Based Simulation Modeling of Construction Fleet Operations Using Multimodal-Process Data Mining. *Journal of Construction Engineering and Management,* 139**,** 04013021.

Akhavian, R. & Behzadan, A. H. (2014). Evaluation of queuing systems for knowledge-based simulation of construction processes. *Automation in Construction,* 47**,** 37-49.

Akhavian, R. & Behzadan, A. H. (2015). Construction equipment activity recognition for simulation input modeling using mobile sensors and machine learning classifiers. *Advanced Engineering Informatics,* 29**,** 867-877.

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

Akinci, B., Kiziltas, S., Ergen, E., Karaesmen, I. Z. & Keceli, F. (2006). Modeling and analyzing the impact of technology on data capture and transfer processes at construction sites: A case study. *Journal of Construction Engineering and Management-Asce,* 132**,** 1148-1157.

Banks, J. (2005). *Discrete Event System Simulation,* New Jersey, Pearson Education.

Bao, L. & Intille, S. S. (2004). Activity recognition from user-annotated acceleration data. *Pervasive computing.* Springer.

Brezmes, T., Gorricho, J.-L. & Cotrina, J. (2009). Activity recognition from accelerometer data on a mobile phone. *Distributed computing, artificial intelligence, bioinformatics, soft computing, and ambient assisted living.* Springer.

Casale, P., Pujol, O. & Radeva, P. (2011). Human activity recognition from accelerometer data using a wearable device. *Pattern Recognition and Image Analysis.* Springer.

Favela, J., Tentori, M., Castro, L. A., Gonzalez, V. M., Moran, E. B. & Martínez-García, A. I. (2007). Activity recognition for context-aware hospital applications: issues and opportunities for the deployment of pervasive networks. *Mobile Networks and Applications,* 12**,** 155-171.

Gong, J. & Caldas, C. H. (2010). Computer vision-based video interpretation model for automated productivity analysis of construction operations. *Journal of Computing in Civil Engineering,* 24**,** 252-263.

Halpin, D. W. (1977). CYCLONE - A Method for Modeling Job Site Processes. *ASCE Journal of Construction Division,* 103**,** 489-499.

Halpin, D. W. & Riggs, L. S. (1992). *Planning and analysis of construction operations,* New York, Wiley.

Jog, G. M., Brilakis, I. K. & Angelides, D. C. (2011). Testing in harsh conditions: Tracking resources on construction sites with machine vision. *Automation in construction,* 20**,** 328-337.

Joshua, L. & Varghese, K. (2010). Accelerometer-based activity recognition in construction. *Journal of Computing in Civil Engineering,* 25**,** 370-379.

Leite, F., Cho, Y., Behzadan, A. H., Lee, S., Choe, S., Fang, Y., Akhavian, R. & Hwang, S. (2016). Visualization, Information Modeling, and Simulation: Grand Challenges in the Construction Industry. *Journal of Computing in Civil Engineering***,** 04016035.

Lin, X., Yacoub, S., Burns, J. & Simske, S. (2003). Performance analysis of pattern classifier combination by plurality voting. *Pattern Recognition Letters,* 24**,** 1959-1969.

Martinez, J. & Ioannou, P. G. (Year) Published. General purpose simulation with stroboscope. the 1994 Winter Simulation Conference (WSC), 1994 Lake Buena Vista, FL. Association for Computing Machinery (ACM), 1159-1166.

Martinez, J. C. (1996). *Stroboscope: State and Resource Based Simulation of Construction Processes.* Ph.D. thesis, University of Michigan.

Mathur, N., Aria, S., Adams, T., Ahn, C. & Lee, S. (2015). Automated Cycle Time Measurement and Analysis of Excavator's Loading Operation Using Smart Phone-Embedded IMU Sensors. *Computing in Civil Engineering 2015.*

Ravi, N., Dandekar, N., Mysore, P. & Littman, M. L. (Year) Published. Activity recognition from accelerometer data. AAAI, 2005. 1541-1546.

Rockwellautomation. (2017). *Arena Simulation Software* [Online]. Available: https://www.arenasimulation.com/ [Accessed December 2016].

Shahandashti, S. M., Razavi, S. N., Soibelman, L., Berges, M., Caldas, C. H., Brilakis, I., Teizer, J., Vela, P. A., Haas, C., Garrett, J., Akinci, B. & Zhu, Z. H. (2011). Data-Fusion Approaches and Applications for Construction Engineering. *Journal of Construction Engineering and Management-Asce,* 137**,** 863-869.

Song, L. & Eldin, N. N. (2012). Adaptive Real-Time Tracking and Simulation of Heavy Construction Operations for Look-Ahead Scheduling. *Automation in Construction,* 27**,** 32-39.

Taneja, S., Akinci, B., Garrett, J. H., Soibelman, L., Ergen, E., Pradhan, A., Tang, P. B., Berges, M., Atasoy, G., Liu, X. S., Shahandashti, S. M. & Anil, E. B. (2011). Sensing and Field Data Capture for Construction and Facility Operations. *Journal of Construction Engineering and Management-Asce,* 137**,** 870-881.

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

Yu, C.-H. & Ohlund, B. (2010). Threats to validity of research design. *Retrieved January,* 12**,** 2012.