Stacked Hourglass Networks for Markerless Pose Estimation of Articulated Construction Robots

C. J. Liang[a], K. M. Lundeen[a], W. McGee[b], C. C. Menassa[a], S. Lee[a] and V. R. Kamat[a]

[a]Department of Civil and Environmental Engineering, University of Michigan, USA
[b]Taubman College of Architecture and Urban Planning, University of Michigan, USA
E-mail: cjliang@umich.edu, klundeen@umich.edu, wesmcgee@umich.edu, menassa@umich.edu,
shdpm@umich.edu, vkamat@umich.edu

Abstract –

The objective of this research is to evaluate vision-based pose estimation methods for on-site construction robots. The prospect of human-robot collaborative work on construction sites introduces new workplace hazards that must be mitigated to ensure safety. Human workers working on tasks alongside construction robots must perceive the interaction to be safe to ensure team identification and trust. Detecting the robot pose in real-time is thus a key requirement in order to inform the workers and to enable autonomous operation. Vision-based (marker-less, marker-based) and sensor-based (IMU, UWB) are two of the main methods for estimating robot pose. The marker-based and sensor-based methods require some additional preinstalled sensors or markers, whereas the marker-less method only requires an on-site camera system, which is common on modern construction sites. In this research, we develop a marker-less pose estimation system, which is based on a convolutional neural network (CNN) human pose estimation algorithm: stacked hourglass networks. The system is trained with image data collected from a factory setup environment and labels of excavator pose. We use a KUKA robot arm with a bucket mounted on the end-effector to represent a robotic excavator in our experiment. We evaluate the marker-less method and compare the result with the robot's ground truth pose. The preliminary results show that the marker-less method is capable of estimating the pose of the excavator based on a state-of-the-art human pose estimation algorithm.

Keywords –

Pose Estimation; Stacked Hourglass; Excavator

1 Introduction

Due to the hazardous working environment, construction site has a higher rate of fatalities and injuries throughout the industry [1]. On average, 53% of the fatal accidents that happen on construction sites are either struck by vehicle or equipment overturns and collisions [2], which causes almost $13 billion in extra cost per year [3]. Blind spots around the equipment are the main cause of such accidents [4]. When workers need to interact with the equipment on job sites, the equipment operator sometimes cannot locate all workers nearby and the workers also cannot locate the equipment components clearly. The prospect of collaborative human-robot teams on construction sites further heightens these concerns and highlights a need for developing on-site articulated equipment pose estimation methods. The pose of the construction equipment, such as an excavator, can be described as the angle between each component and the 6 degree-of-freedom (6 DOF) coordinates. Therefore, determining each joint location and the angle between each component is the primary goal of the machine pose estimation, as shown in Figure 1.

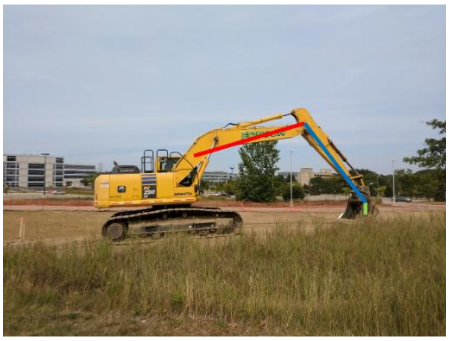

Figure 1. Excavator pose is determined by identifying its joints and components.

In the real practice, two types of pose estimation methods are used on construction equipment, namely non-visual sensor-based and vision-based pose estimation method. For sensor-based pose estimation methods, Inertial Measurement Unit (IMU), Global

Positioning System (GPS), Wireless Local Area Network (WLAN), Radio Frequency Identification (RFID), and Ultra-Wide Band (UWB) are mainly deployed on equipment and construction site. IMU sensors need to be mounted on excavator joint components to measure the angle [5], which has drift issues [6]. GPS is known for outdoor used only [7], which is not suitable for some indoor construction site. WLAN system requires significant amounts of effort for calibration [8]. RFID and UWB methods both require sufficient preinstalled tags and readers on equipment and infrastructure [9–11]. They generally suffer from missing data issues [12] and are inadequate for pose estimation [13]. In addition, most of these methods cannot provide orientation information directly, except for IMU, are not suitable for construction scenarios.

On the other hand, vision-based pose estimation methods are capable of analyzing position information as well as orientation information directly from input data, such as videos or point clouds. These methods generally recognize construction equipment on site [14–17], then estimate their 6 degrees-of-freedom (6 DOF) pose [18,19], which can be categorized to two different group: marker-based and marker-less pose estimation. The marker-based pose estimation method recognizes all the markers mounted on equipment and estimates the pose by their geometric relations [20,21], whereas the marker-less pose estimation method directly extracts image features and estimates the pose by them [18]. The marker-based method has been extensively applied in indoor localization and facility management [22,23]. Similar to sensor-based pose estimation method, they also require preinstalled markers on equipment and environment.

In addition to the marker-based method, the marker-less pose estimation method only requires an on-site camera system, which is common on modern construction sites. Feature descriptor based is the first type of marker-less pose estimation method, such as Histograms of Oriented Gradient (HOG) [16], 3D principal axes descriptor (PAD) [14], Iterative Closest Point (ICP) [24], or Viewpoint Feature Histogram (VFH) [18]. Convolutional Neural Networks (CNN) is another type of pose estimation method [25], which has higher performance (accuracy and speed) in comparison with all other vision-based methods, especially for human pose estimation. Therefore, in this study, a CNN based marker-less pose estimation system is presented, which can distinguish excavator joint components and estimate their poses in images. This system is built on a state-of-the-art human pose estimation network [26,27] and trained on an excavator image dataset collected in a factory setup lab environment. The excavator pose in this research is defined as the boom, stick, and bucket pixel-wise 2D location.

2 Marker-less Pose Estimation System

Our marker-less pose estimation system is developed based on a state-of-the-art human pose estimation algorithm, namely stacked hourglass network by Newell et al. [26,27]. This network scales the training image into different resolution and captures features, then combines the information together to predict the pose. Compared with the human pose, the construction equipment pose is much simpler, thus requires less information across different image resolutions. The detailed network architecture is further discussed in the next section.

2.1 System Network Architecture

We modify the stacked hourglass network to fit our target construction machine, mainly excavator. Unlike the complicated human skeleton, excavator pose only requires identifying three components, which are bucket, stick, and boom, as shown in Figure 1. Therefore, the complexity of the network needed is much less than the original network. Figure 2 shows the network architecture. Two convolutional layers followed by a max pooling layer are first applied to the training image, which shrinks the image down to the size of 64 pixels. Then three subsequent convolutional layers upscale the image to the size of 256 pixels before the hourglass module. Finally, three hourglass modules, output prediction modules, and residual link modules are used in the network. All the convolutional layers are followed by ReLu activation function, with stride 1 except the Conv1 layer with stride 2, and with batch normalization except the convolutional layers in the output prediction module.

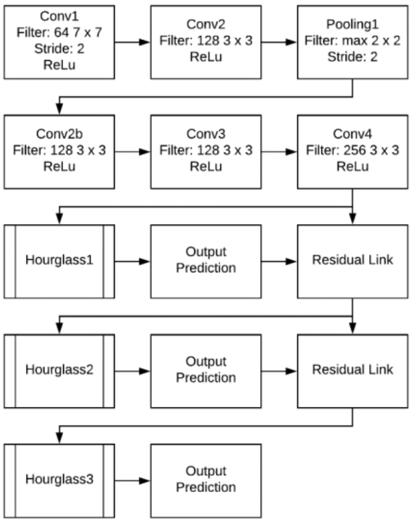

Figure 2. Full system network architecture, 3 hourglass modules are used in our system.

The hourglass module is the main part to collect features across different resolution, which is illustrated in Figure 3. The input passes into two parallel routes. In the first route, only one convolutional layer is applied to upscale the input to the size of 256 pixels. In the second route, one max pooling layer followed by three convolutional layers are applied to downscale the input to the size of 384 pixels, then resized to the size of 256 pixels, as the first route result. Finally, two route results are added together through elementwise summation to generate the output. This can preserve the global feature and capture the local feature as well. In the Hourglass2 and Hourglass3 module, we change the Conv_low2 layer to another hourglass module. This recursive hourglass module will increase the output size for more features.

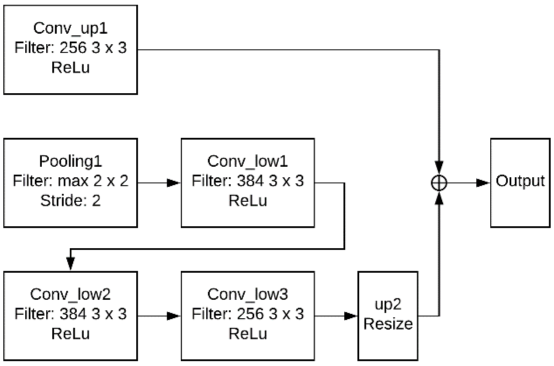

Figure 3. Hourglass module architecture, convolutional upscale and downscale layers are the main features of the hourglass shape.

The output prediction module and residual link module are applied after the hourglass module, as shown in Figure 4. Two convolutional layers are used in the output prediction module to generate the heat map of the possibility of the location of each joint. Figure 5 shows the concept of the prediction heat map. Each red dot represents the highest probability of each joint location from which we can estimate the pose, as shown in Figure 1. The final layer is a one-by-one convolutional layer, which aims to calculate the possibility across the depth of the output of the Conv5 layer. On the other hand, the residual link module combines the output from the previous hourglass and after the output prediction module to generate the input for next hourglass. The repeated hourglass and residual link module can preserve the spatial location and relation of each feature and apply to the final prediction step.

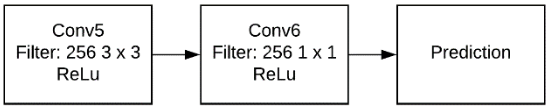

Figure 4. Output prediction layers, the previous hourglass prediction results are added with current output prediction.

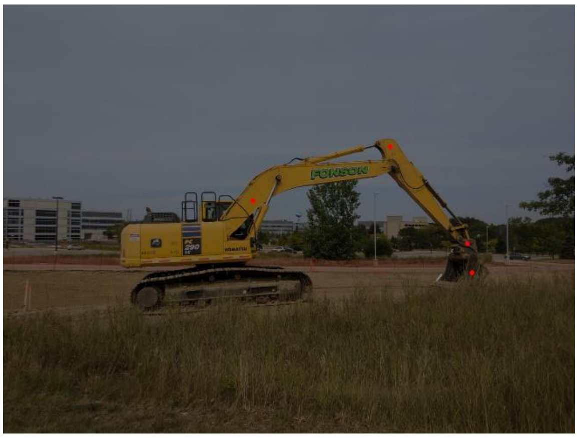

Figure 5. The concept of the prediction heat map. Each red dot represents the highest probability of each joint location.

2.2 Training Details

We use the L_2-norm loss function to train our network, as shown in (1):

$$L_2(\hat{X}_p, X_L) = \sum \left(\hat{X}_p - G(X_L)\right)^2 \tag{1}$$

where $\hat{X}_p$ represents the predicted pose and X_L represents the labeled ground truth training data, $G(\cdot)$ represents the Gaussian kernel function with 1-pixel standard deviation. The loss function directly calculates the error between training and predicting image.

We implement the network system by modifying the original network using PyTorch [28] and the loss function described above. The network is trained on an excavator image dataset, which we collected from a factory setup lab environment with a simulated excavator and real construction site with real excavators. All the hyper-parameters are set the same as in [27]. The excavator dataset contains 1,000 training images and 500 testing images aligned with their pose annotating data. The detailed lab environment setup is discussed in section 3.

3 Experiment

We collect the image data from a factory setup lab environment and from real construction sites. The dataset is separated into training and testing groups. The algorithm is trained by the training group and evaluated by the testing dataset.

3.1 Implementation

We used a KUKA 7 DOF robot arm to simulate the excavator and capture the image of the robot arm with different poses. The upper arm represents the excavator stick and the lower arm represents the excavator boom. A bucket is mounted on the robot arm for a more realistic simulation. Figure 6 shows the simulated excavator in a factory setup lab environment. In order to control the robot as an excavator, the profile of the mounted bucket must remain perpendicular to the ground level. We controlled the robot arm to perform several excavator tasks such as digging, moving, or unloading.

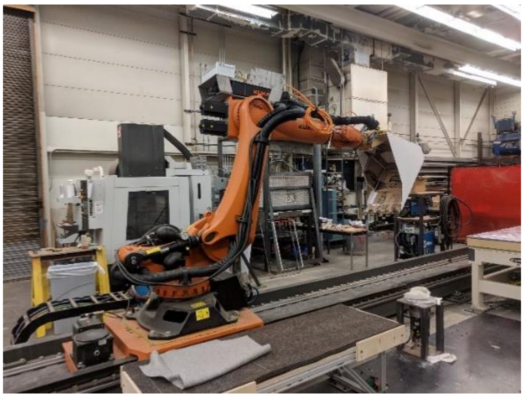

Figure 6. The simulated excavator by a robot arm mounted with an excavator bucket.

We used a Point Grey camera to capture the image of the simulated excavator. The camera was deployed in 5 different location and orientation near the excavator to increase the variety of the dataset. A total of 1,000 images were collected; 750 of them were used as training images and 250 of them were used as testing images. Figure 7 shows an example of the simulated excavator dataset with different camera location and orientation. The joints of the simulated excavator were labeled in 2D pixel-wise location via MATLAB code. The structure of the annotation data is the same as the well-known human pose dataset (MPII) [25].

To increase the variety of the dataset and augmented the background of the dataset, we also collected image data from the real construction site with real excavators, as shown in Figure 1. A total of 500 images were collected; 250 of them were used as training images and 250 of them were used as testing images.

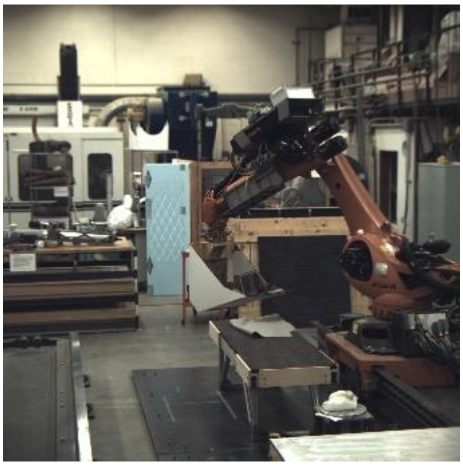

Figure 7. Example of the simulated excavator dataset with different camera location and orientation.

4 Results

We evaluate the proposed method by comparing the prediction results of the testing images and the ground truth. Figure 8 demonstrates the results of the excavator pose estimation. The green, blue, and red line are corresponding to the bucket, stick, and boom prediction. These two images are in the testing dataset. We also evaluate the Euclidean distance between the predicted joint location and the ground truth joint location, and the error percentage of the predicted component length and the ground truth, which can be seen in Table 1 and For the error percentage of the predicted component length and the ground truth, we only evaluated the lab dataset because the length of each robot arm component is known but the real site excavators are unknown. The result is shown in Table 2. The error percentage of the boom and stick is about 35% to 45%, and the bucket is 62%. The reason for the high error percentage in the bucket case is the occlusion issue. When the bucket is blocked or out of range, the predicted bucket location will be far away from its true location. In addition, the ground truth length of the bucket is short, which increases the differences between the ground truth and the false predicted result. Figure 9 shows two results of false prediction caused by occlusion.

Table 2. The average Euclidean distance between the lab testing dataset and ground truth is 50.05 pixels and between the real site testing dataset and the ground truth is 71.95 pixels. The bucket location has the highest error because the bucket is blocked (occluded) or out of range in some of the image. The model still tries to find the location in these cases, which increases the error

distance. The error in the real site dataset is higher than the lab dataset. This is because the real site dataset has a greater variety of excavators and backgrounds. Only some of these variations were included in the testing dataset, so this caused a decrease in accuracy.

Table 1. Results of the average Euclidean distance (pixel-wise) between the predicted and the ground truth joint location.

(pixels)	Lab Dataset	Real site dataset
Boom	42.01	67.12
Boom Stick	45.37	59.99
Stick Bucket	44.68	65.67
Bucket	68.13	95.03

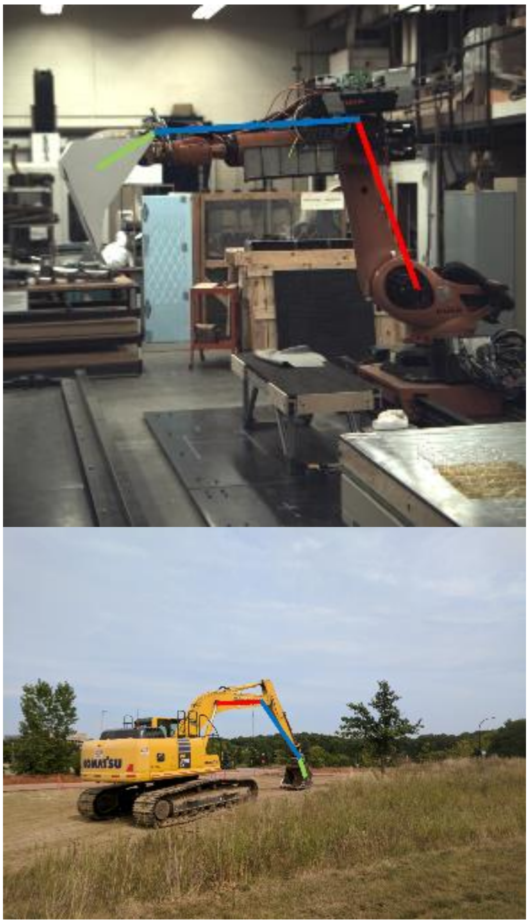

Figure 8. The result of the excavator pose estimation. On the top is the simulated excavator and on the bottom is the real excavator.

For the error percentage of the predicted component length and the ground truth, we only evaluated the lab dataset because the length of each robot arm component is known but the real site excavators are unknown. The result is shown in Table 2. The error percentage of the boom and stick is about 35% to 45%, and the bucket is 62%. The reason for the high error percentage in the bucket case is the occlusion issue. When the bucket is blocked or out of range, the predicted bucket location will be far away from its true location. In addition, the ground truth length of the bucket is short, which increases the differences between the ground truth and the false predicted result. Figure 9 shows two results of false prediction caused by occlusion.

Table 2. Results of the error percentage of the predicted component length and the ground truth.

(%)	Lab Dataset
Boom	46.8
Stick	34.3
Bucket	62.8

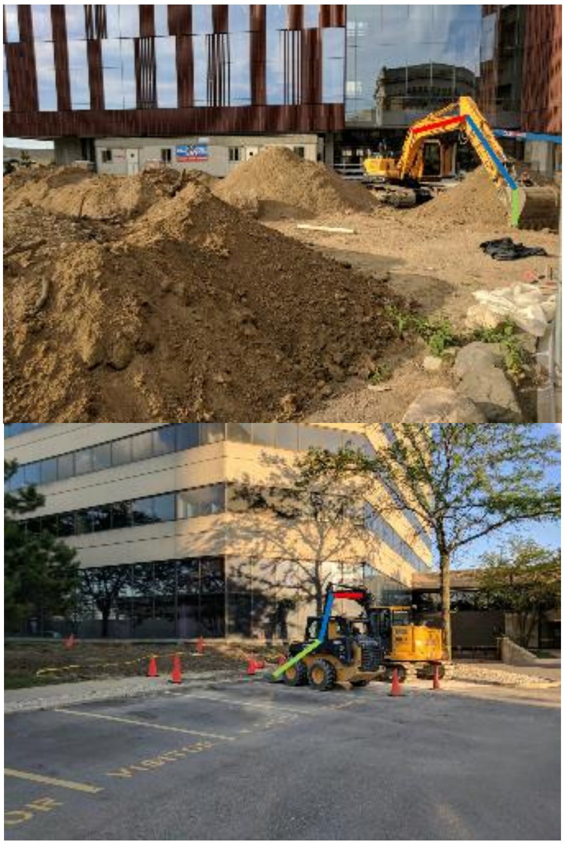

Figure 9. The result of the false prediction, both are out of range (top) or blocked (bottom).

Based on the evaluation results, occlusion is the primary issue of the proposed system, which can be tackled by increasing the number and variety of the training dataset. Another problem is the multiple excavator situation. The proposed system can only identify one excavator pose. If there are two or more excavators in the image, the result will fail. We will design a new network or method for multiple excavator situation in the future work.

5 Conclusion

In this research, we proposed and evaluated a vision-based marker-less pose estimation system for construction robots, for which we used an excavator as our test-bed. The excavator boom, stick, and bucket joint positions are estimated with pixel-wise coordinates. We adapted and modified the state-of-the-art human pose estimation convolutional network, i.e., the stacked hourglass network, for our application. Three stacked hourglass modules and two residual links are included in the network. The network model is trained on an excavator dataset, which we collected and annotated from a factory setup lab environment with a KUKA robot arm representing an excavator from a real construction site. The results showed that the system can estimate the boom and stick joints but had higher estimation error for the bucket location due to the occlusion issue. Therefore, for the future work, more training image data with higher variety will be collected. We will also modify the proposed network to adapt to the multiple excavator situation. The comparison between the proposed system and the sensor-based system will be conducted as well.

6 Acknowledgments

The authors would like to acknowledge the financial support for this research received from the US National Science Foundation (NSF) via grant 1734266. Any opinions and findings in this paper are those of the authors and do not necessarily represent those of the NSF.

References

[1] Zhou Z., Goh Y.M. and Li Q. Overview and analysis of safety management studies in the construction industry. *Safety Science*. 72: 337–350, 2015.

[2] BLS. An analysis of fatal occupational injuries at road construction sites, 2003–2010. *Monthly Labor Review*, 2013.

[3] CPWR. *The construction chart book: the U.S. construction industry and its workers*, 4th ed. CPWR - The Center for Construction Research and Training, Silver Spring, MD, 2008.

[4] Teizer J., Allread B.S. and Mantripragada U. Automating the blind spot measurement of construction equipment. *Automation in Construction*, 19(4): 56–64, 2010.

[5] Bender F.A., Göltz S., Bräunl T. and Sawodny O. Modeling and offset-free model predictive control of a hydraulic mini excavator. *IEEE Transactions on Automation Science and Engineering*, 14(4): 1682–1694, 2017.

[6] Park J., Chen J. and Cho Y.K. Self-corrective knowledge-based hybrid tracking system using BIM and multimodal sensors. *Advanced Engineering Informatics*, 32: 126–138, 2017.

[7] Groves P.D. Shadow matching: a new GNSS positioning technique for urban canyons. *The Journal of Navigation*, 64(3): 417–430, 2011.

[8] Aziz Z., Anumba C.J., Ruikar D., Carrillo P.M. and Bouchlaghem N.M. Context aware information delivery for on-site construction operations. In *Proceedings of the CIB-W78 International Conference on Information Technology in Construction*, 321–332, Dresden, Germany, 2005.

[9] Teizer J., Venugopal M. and Walia A. Ultrawideband for automated real-time three-dimensional location sensing for workforce, equipment, and material positioning and tracking. *Transportation Research Record: Journal of the Transportation Research Board*, 2081: 56–64, 2008.

[10] Khoury H.M. and Kamat V.R. Evaluation of position tracking technologies for user localization in indoor construction environments. *Automation in Construction*, 18(4): 444–457, 2009.

[11] Jo B.-W., Lee Y.-S., Kim J.-H., Kim D.-K. and Choi P.-H. Proximity warning and excavator control system for prevention of collision accidents. *Sustainability*, 9(8): 1488, 2017.

[12] Vahdatikhaki F., Hammad A. and Siddiqui H. Optimization-based excavator pose estimation using real-time location systems. *Automation in Construction*, 56: 76–92, 2015.

[13] Chai J., Wu C., Zhao C., Chi H.-L., Wang X., Ling B.W.-K. and Teo K.L. Reference tag supported RFID tracking using robust support vector regression and Kalman filter. *Advanced Engineering Informatics*, 32: 1–10, 2017.

[14] Chen J., Fang Y., Cho Y.K. and Kim C., Principal axes descriptor for automated construction-equipment classification from point clouds. *Journal of Computing in Civil Engineering*, 31(2): 04016058, 2017.

[15] Soltani M.M., Zhu Z. and Hammad A. Automated annotation for visual recognition of construction resources using synthetic images, *Automation in Construction*, 62: 14–23, 2016.

[16] Rezazadeh Azar E., Dickinson S. and McCabe B. Server-customer interaction tracker: computer vision–based system to estimate dirt-loading cycles. *Journal of Construction Engineering and*

Management, 139(7): 785–794, 2013.

[17] Rezazadeh Azar E. and McCabe B. Automated visual recognition of dump trucks in construction videos. *Journal of Computing in Civil Engineering*, 26(6): 769–781, 2012.

[18] Liang C.-J., Kamat V.R. and Menassa C.M. Real-time construction site layout and equipment monitoring. In *Proceedings of the 2018 Construction Research Congress*, New Orleans, LA, 2018.

[19] Soltani M.M., Zhu Z. and Hammad A. Skeleton estimation of excavator by detecting its parts. *Automation in Construction*, 82: 1–15, 2017.

[20] Lundeen K.M., Dong S., Fredricks N., Akula M., Seo J., Kamat V.R. Optical marker-based end effector pose estimation for articulated excavators. *Automation in Construction*, 65: 51–64, 2016.

[21] Rezazadeh Azar E., Feng C. and Kamat V.R. Feasibility of in-plane articulation monitoring of excavator arm using planar marker tracking. *Journal of Information Technology in Construction (ITcon)*, 20(15): 213–229, 2015.

[22] Xu L., Kamat V.R. and Menassa C.C. Automatic extraction of 1D barcodes from video scans for drone-assisted inventory management in warehousing applications. *International Journal of Logistics Research and Applications*, 1–16, 2017.

[23] Feng C. and Kamat V.R. Plane registration leveraged by global constraints for context-aware AEC applications. *Computer-Aided Civil and Infrastructure Engineering*, 28(5): 325–343, 2013.

[24] Lundeen K.M., Kamat V.R., Menassa C.C. and McGee W. Scene understanding for adaptive manipulation in robotized construction work. *Automation in Construction*, 82: 16–30, 2017.

[25] Andriluka M., Pishchulin L., Gehler P. and Schiele B. 2D human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3686–3693, Columbus, OH, 2014.

[26] Newell A., Huang Z. and Deng J. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*, 2274–2284, Long Beach, CA, 2017.

[27] Newell A., Yang K. and Deng J. Stacked hourglass networks for human pose estimation. In *Computer Vision – ECCV 2016*, 483–499, Amsterdam, Netherlands, 2016.

[28] Paszke A., Gross S., Chintala S., Chanan G., Yang E., DeVito Z., Lin Z., Desmaison A., Antiga L. and Lerer A. Automatic differentiation in PyTorch. In *Advances in Neural Information Processing Systems*, Long Beach, CA, 2017.