Single-Stage Spatiotemporal Activity Recognition of Excavators: A Case Study

Ali Ghelmani[1], Ghazaleh Torabi[2], Amin Hammad[1]*, Chen Chen[3]

[1]Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada
[2]ZestyAI, Montreal, Canada
[3]School of Civil Engineering and Architecture, Zhejiang University of Science and Technology, Hangzhou, China
ali.ghelmanirashidabad@concordia.ca, ghazalehtrb@gmail.com, amin.hammad@concordia.ca*,
chen.c@zust.edu.cn

Abstract -

Manual excavator activity monitoring to evaluate their performance and productivity is laborious, time-consuming, and error-prone. To address these problems, many automated computer vision-based frameworks have been developed for the detection of excavators and the classification of their activities. Most of the current methods consist of several separately optimized modules that are applied to the input video sequentially. Recently, single-stage spatiotemporal activity recognition methods are gaining more popularity in the construction community. You Only Watch Once (YOWO) network and its variation (i.e., YOWO53) have proved to be superior to the three-stage approaches for activity recognition of construction workers. This paper investigates the benefits of using YOWO and YOWO53 over the three-stage methods for the activity recognition of excavators, by utilizing a large custom dataset of 1,060 video clips collected from both local construction sites and YouTube, with different camera angles, illuminations, occlusions, weather conditions, and video resolutions. The results demonstrate 88.9 and 88.7% classification accuracy and F1-score, respectively for the YOWO method compared to 70.4% and 69.8% classification accuracy and F1-score for the three-stage method. This indicates the feasibility and benefits of deploying the single-stage methods to near real-time applications.

Keywords –

Site monitoring, Activity recognition, Computer vision

1 Introduction

Nowadays, the growing demand for the completion of construction projects within schedule and under budget has resulted in the development of automated, continuous monitoring routines to provide project managers with vital productivity and safety information [2]. Traditionally, monitoring the activities of excavators and other earthmoving equipment is performed manually by superintendents on the site. However, such methods can be very time-consuming, labor-intensive, and error-prone especially on large construction sites [1, 2]. Considering that excavators are at the core of earthmoving operations [3], monitoring their activities can provide information about work cycle duration and consequently productivity. This information in turn enables site managers to make more informed project-related decisions, such as adjusting resource allocations. Considering that excavators are at the core of earthmoving operations [3], monitoring their activities can provide productivity and work cycle duration information, which in turn enables site managers to make more informed project-related decisions, such as resource allocations and scheduling [4, 5]. Videos can provide detailed information about the visual features and physical motions of equipment, and therefore increase the interpretability of the results and their shortcomings by viewing the recorded video and the detected activities [2, 6]. Before the rise of deep learning, vision-based methods generally relied on hand-crafted features to extract useful information for activity recognition from images and videos [7]. However, advances in deep learning methods demonstrated their superiority over traditional hand-crafted methods in different applications such as object detection [8] and activity recognition [9], which resulted in a corresponding change in the use of vision-based methods in the construction domain.

Convolutional Neural Networks (CNN) are the main building blocks in all vision-based deep learning methods, and in the past few years, many 2D CNN-based construction equipment activity recognition methods have been proposed. For instance, Roberts et al. [1] used a combination of 2D CNN with Hidden Markov Models to detect, track, and identify the activities of excavators and dump trucks. Luo et al. [10] used a combination of 2D CNNs and relevance networks for detecting various construction-related objects and their associated set of interactive activities by exploiting the two-dimensional pixel proximity of the detected objects. Kim and Chi [11] also performed interaction analysis to identify the activities and operation cycles of excavators and dump trucks by combining 2D CNN and Long Short-Term Memory (LSTM) architectures. Similar combinations of 2D CNNs and LSTMs, were also used by Slaton et al. [12] to detect the routine tasks of excavators and roller compactors, and by Kim et al. [6] to detect excavator activities via exploiting their sequential working patterns for automatic productivity analysis.

While 2D CNN-based methods try to combine the spatial and temporal information using different methods, 3D CNN-based methods incorporate the spatiotemporal data extraction into a single architecture, which allows the deep learning models to extract relevant spatiotemporal data. Chen et al. [2, 13] proposed a three-stage method in which excavators are first detected in the input frames. Then, the detected excavators are fed into a tracking algorithm, and finally, the tracked results are input to a 3D CNN network to classify the activities. A similar three-stage framework was also proposed by Lou et al. [14] in which workers were first detected using the You Only Look Once (YOLOv3) network. The detected workers were then tracked, and the activities performed by them were classified using a 3D CNN architecture. Although these frameworks can potentially extract more informative spatiotemporal features using 3D CNN architectures, their three-stage approach still limits their accuracy. The main limitations of three-stage methods are: (1) not being fully optimized, and (2) the propagation of errors from earlier stages to the later ones, which results in the degradation of the performance of the entire framework [14,15]. For example, if an equipment is not detected in a few frames or if it is not tracked properly through the frames in which an activity is occurring, the final 3D CNN stage cannot correctly classify the underlying activity.

The benefits of using a single-stage method over the three-stage methods were studied for the case of detecting activities of construction workers by Torabi et al. [16]. They proposed a network called You Only Watch Once 53 (YOWO53) to jointly detect construction workers that appeared small in the video frames and classify their activities. YOWO53 is based on a general human activity recognition network called YOWO [17]. The results showed YOWO53 improves the detection recall of YOWO for small objects (e.g., workers) by at least 2%, and both single-stage networks (i.e., YOWO and YOWO53) improved the activity classification accuracy of one of the state-of-the-art three-stage methods [2] by at least 16%. Jung et al. [15] also proposed a single-stage architecture for detecting the activities of multiple construction equipment simultaneously. This framework uses a 3D CNN architecture and performs equipment detection and activity recognition in one stage to alleviate the limitations of the three-stage methods. However, 39% of the video clips in the reported dataset of seven activities correspond to the idling state of the equipment. Such a dataset, in addition to being unbalanced, limits the real-world applicability of the trained model.

The aim of this paper is to investigate whether the same improvement achieved by single-stage YOWO53 method for workers [16] is achievable for the case of excavators. Furthermore, another important factor in the final performance and applicability of a developed activity recognition model is the size and variability of the data included in a dataset. To this end, a large balanced dataset of excavator activities with more than 1,060 video clips, collected both from local construction sites and YouTube, has been gathered. The prepared dataset contains the three common excavator activities of digging, swinging, and loading the trucks under various camera angles, illuminations, occlusions, weather conditions, and video resolutions. Thus, enabling a thorough evaluation of the YOWO and YOWO53 methods under various real-world conditions.

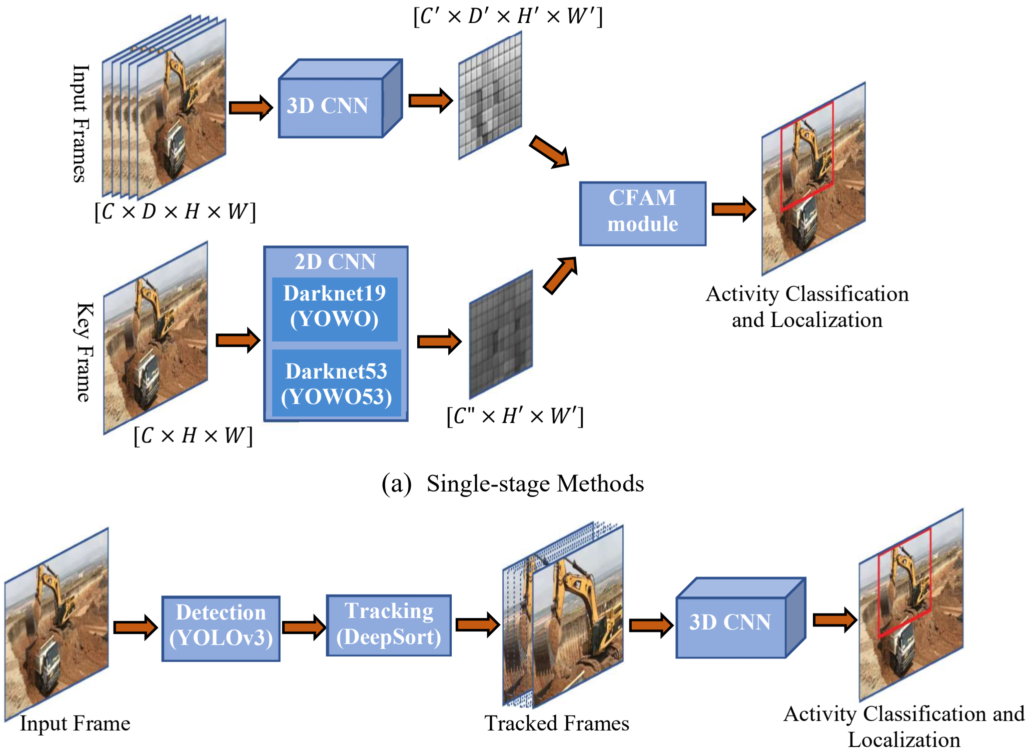
2 Methods Used in The Case Study

The single-stage YOWO [17] and YOWO53 [16] methods are compared in this study with a state-of-the-art three-stage method proposed by Chen et al. [2]. The general architectures of the selected methods are shown in Figure 1, and a more detailed description of these methods is presented in the following sections.

2.1 YOWO

YOWO [17] is a spatiotemporal activity recognition method, which uses two branches in its architecture. One branch extracts 2D features from the current frame while the second branch extracts 3D features from a stack of successive frames. Afterwards, the outputs of the two branches are combined using a channel fusion and attention mechanism (CFAM), which provides the essential performance boost.

The 3D CNN branch is utilized for extracting the spatiotemporal features. In this work, the ShuffleNetV2_2x [18] 3D CNN is chosen for this branch for comparison with the activity recognition results of workers [16]. The input to this network is a video clip comprised of a sequence of frames with the dimension of $[C \times D \times H \times W]$, with C being equal to 3 (RGB channels), D representing the number of input frames, and H and w representing the height and width of the


Input Frames
$[C \times D \times H \times W]$
3D CNN
$[C' \times D' \times H' \times W']$
CFAM module
Activity Classification and Localization
Key Frame
$[C \times H \times W]$
2D CNN
Darknet19 (YOWO)
Darknet53 (YOWO53)
$[C" \times H' \times W']$
Input Frame
Detection (YOLOv3)
Tracking (DeepSort)
Tracked Frames
3D CNN
Activity Classification and Localization

(a) Single-stage Methods

(b) Three-stage Method

Figure 1. The general architecture of the selected methods

frames, respectively. The shape of the output is $[C' \times D' \times H' \times W']$, with C' being the number of output channels, $D' = 1$, $H' = H/32$, and $W' = W/32$. By default, the output of the 3D CNN branch is 4-dimensional, while the output of the 2D CNN branch is 3-dimensional. To make the output of these two branches compatible before combining, the output of the 3D CNN branch is designed to have a reduced depth component ($D' = 1$), which can be dropped, hence becoming three-dimensional in effect.

While some studies only rely on a single 3D CNN network for simultaneous activity recognition and localization (e.g., [15]), the YOWO model also uses a 2D CNN branch in parallel to provide more accurate spatial information. YOWO uses the Darknet19 [19] network in the 2D CNN branch for object detection, which takes images of the form $[C \times H \times W]$ as input, while the shape of the output feature map is of the form of $[C^{"} \times H' \times W']$, where $C"$ is the number of output channels.

Afterwards, the output of the two branches is input into the CFAM module, which fuses the two 2D CNN and 3D CNN extracted information. Since the two branches are designed to have the same shape for their output feature maps, they can be easily concatenated along the channel dimension. The CFAM module then utilizes the fused feature maps to generate a combination of both motion and appearance data. Finally, the loss function used to train the YOWO model, is inspired by the losses defined in YOLO [19] and Fast R-CNN [20] models, and is comprised of the activity classification loss, and the localization loss between the bounding box predicted by the model and the ground truth bounding box.

2.2 YOWO53

The YOWO method has a modular architecture in which the networks in the 2D CNN branch and the 3D CNN branch can be replaced by other corresponding networks. The YOWO53 model exploits this architecture by applying the following modifications. As stated in Section 2.1, the YOWO method uses the Darknet19 network in the 2D CNN branch for extracting spatial features from the current frame. While the Darknet19 network is fast, its detection accuracy for small objects is not very high. To address this problem, the Darknet53 network [21] is utilized in the YOWO53 method, which although slower, provides more accuracy for detecting small objects and hence is more suitable for applications in the construction domain. Particularly, considering that most of the surveillance cameras on construction sites are installed at high altitudes and consequently, workers and construction equipment at a distance can appear very small in the recorded site videos. Following the above change in the 2D CNN branch, to keep the output shape of the 2D and the 3D branches consistent for fusion in CFAM, a single max-pooling layer is removed from the architecture of the network used in the 3D branch. Thus, doubling the size of the output feature maps of the two branches ($H' = H/16$, and $W' = W/16$). This modification not only allows the concatenation of the two feature maps but also decreases the receptive field of YOWO53, which helps with the detection of small objects.

The receptive field of a particular feature in the output feature map of a CNN is the region in the input image that this feature encodes. The size of this region depends on the depth of the feature as well as the combination of the previous layers. Usually as the size of the output feature map is reduced, its receptive field increases. For example, if an image is reduced to a single feature by a CNN, this feature encodes the important information of the entire input image. In object detection, the size of the object that can be detected by the network depends on the receptive field of the last layer (detection layer). If the size of the object is larger than the receptive field of a layer, it may not be correctly detected using the output feature map of that layer. Larger feature maps have smaller receptive fields and can be used to detect smaller objects.

2.3 Three-stage method

To investigate the benefits of using single-stage methods over the three-stage method for the activity recognition of excavators, YOWO and YOWO53 methods are compared with the state-of-the-art three-stage method proposed by Chen et al. [2]. This method is composed of detection using the YOLOv3 method, tracking using the Simple Online and Real-time Tracking (Deep SORT) method [22], and activity recognition using the 3D-ResNext-101 [23] network, with each stage optimized separately. The previous studies (e.g., [2,15]) did not fine-tune the Deep SORT module since it is one of the state-of-the-art methods capable of tracking multiple objects at the same time. The detection module can be trained using simple frame-level bounding box annotations. However, the activity recognition module used in the three-stage method requires the input video clip to contain only a single excavator performing a single activity. Thus, the detected and tracked excavators should be cropped before being input into the 3D CNN network for activity recognition.

3 Dataset description

Considering that the majority of publicly available datasets for the task of activity recognition are focused on human activities in various environments [24, 25], the first step in excavator activity recognition is to create a proper dataset. The video clips used in creating the dataset were manually collected from various sources including local construction sites and videos posted online on websites such as YouTube. Each video clip contains one or more excavators performing three types of activities: digging, swinging, and loading the trucks. To add to the diversity of the collected dataset and enable a thorough analysis of the selected methods, the videos are collected from 25 different construction sites, incorporating various site conditions, such as different camera angles, illuminations, occlusions, weather conditions, and video resolutions. The detailed statistics of the collected dataset are presented in Table 3. For labeling the collected dataset, the Computer Vision Annotation Toolbox (CVAT) [35], which is a free web-based video and image annotation toolbox [35], was used in this paper. The ground truth for each labeled frame includes the type of the occurring activity and the top left and bottom right coordinates of the encompassing bounding box.

4 Implementation details

All of the models are trained on three RTX A6000 GPUs in Ubuntu 20.04 and Python 3.7 environment and PyTorch 1.8, with 80% of the video clips randomly selected for training and the remaining 20% used for testing. The ShuffleNetV2_2x 3D CNN network was pre-trained on the large-scale Kinetics-600 [26] dataset. Only the last layer of this network is fine-tuned on the collected excavator dataset. The 2D CNN networks, i.e., Darknet19 and Darknet53, are pre-trained on the COCO [27] dataset and only their last two layers are fine-tuned in this work. The models are trained for 20 epochs using the Adam optimizer [28] and their best results are saved.

5 Experimental results

Table 1 provides a comparative performance of the YOWO, YOWO53, and the three-stage methods. It can be seen that both YOWO and YOWO53 methods significantly outperform the three-stage method. The results in Table 1 show that the YOWO53 method obtains 15% improvement in classification accuracy and 15.6% improvement in F1-score over the three-stage method, which aligns closely with the results reported in [16]. However, in contrast to the performance improvements of YOWO53 over YOWO for worker activity recognition [16], Table 1 shows that the YOWO method obtains 3.5% improvement in classification accuracy and 3.3% improvement in F1-score over the YOWO53 method for excavator activity recognition.

Table 1. Comparing YOWO, YOWO53, and the three-stage method

	Classification accuracy (%)	**F1-score (%)**
YOWO	88.9	88.7
YOWO53	85.4	85.4
Three-stage method	70.4	69.8

To further examine the difference in the performance of the YOWO and YOWO53 methods, Table 2 shows the classification accuracy, localization recall, overall precision, overall recall, and F1-score obtained by training and evaluating YOWO and YOWO53 methods on three different input sizes. The results for the smallest frame size (i.e., 128×128) agree with what was reported for workers in [16], with YOWO53 achieving better performance than YOWO in all of the metrics. However, as the size of the input frame increases, the performance of YOWO53 drops lower than YOWO. Additionally, after the second smallest input size (i.e., 256×256), the performance of YOWO53 drops by around 1% in all metrics. This indicates that the smaller receptive field of the YOWO53 method (as stated in Section 2.2), while efficient for detecting workers, is not capable of covering big equipment such as excavators and consequently, adversely affects the model performance. Finally, the overall best result is obtained by the YOWO method using the largest input size (i.e., 448×448), with 88.9% classification accuracy, and 88.7% F1-score.

To evaluate the real-world applicability of the YOWO and YOWO53 methods, the number of parameters of both methods, as well as the speed of processing each input size along with the highest batch size that can fit in the three RTX A6000 GPUs are presented in Table 2. It can be seen that the YOWO method processes 448×448 frames at 10.7 FPS, which is comparable to the processing speed of the YOWO53 methods for the same frame size. However, it should be noted that YOWO is a smaller network compared to YOWO53 with 79 million parameters compared to 90 million parameters, allowing higher batches to be processed at the same time by the network.

5.1 Sensitivity analysis

To further investigate the performance of the best YOWO model under various conditions, a sensitivity analysis was performed with different video conditions such as camera angles, illumination, occlusion, weather conditions, and video resolution. The results are presented in Table 3, demonstrating the high performance and applicability of the model for various real-world conditions.

As mentioned in Section 4, 80% of the total number of video clips in the dataset was selected randomly for training, while the remaining 20% was used for testing the performance of the model. A consequence of this division strategy is the varying ratio between training and testing data for each of the considered sensitivity analysis cases, especially for cases in which the total amount of data in the dataset is relatively small. For example, for the high occlusion category there are 455 frames for testing compared to 1,493 frames for training, a ratio of almost 1:3, while for the snowfield category, the amount of training data is even less than that of testing, with 421 frames available for training while 780 frames are used for testing. However, it should be noted that even in the snowfield category with such a train/test data imbalance, the model still achieves 94.7% activity classification accuracy and recall, showing its high accuracy even for cases with low training data, as long as the image quality is not severely degraded (e.g., high occlusion category).

Another interesting example is the "Below ground level" category, with only 143 frames from one video clip for testing, for which due to the high quality of the input video clip, the performance is still high (82.6% activity classification accuracy and 82.5% recall). Another effect of the low amount of training data for some cases can be seen in the apparent performance contradiction for the low-resolution video clips only with 13,056 training and 3,226 testing frames, which seems to outperform the high-resolution cases with 117,173 training and 29,840 testing frames (about nine times more frames). However, after further investigation, the lower performance for high resolution video clips is found to be due to the inclusion of most other difficult cases, which resulted in almost the same performance for this category as the full dataset reported in Table 2.

Finally, it can be seen from Table 3 that the worst performance of the model (66.4% activity classification accuracy) is in the high occlusion category, which is generally one of the biggest limitations of single-camera CV-based methods. However, considering that in these cases more than half of the excavator is not visible due to occlusion by other equipment, self-occlusion, or not being fully in the camera's field of view, the results show the impressive performance of the model.

Table 2. Performance comparison for variants of YOWO and YOWO53 methods

Model	Input size	Classification accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Speed (FPS)	Batch size	#param
YOWO	448 × 448	88.9	88.7	88.7	88.7	10.7	64	~79M
YOWO	256 × 256	88.0	87.2	87.4	87.3	12	256	
YOWO	128 × 128	84.3	82.5	82.8	82.7	12.6	256	
YOWO53	448 × 448	85.4	85.4	85.4	85.4	10.4	32	~90M
YOWO53	256 × 256	86.8	86.3	86.3	86.3	10.9	32	
YOWO53	128 × 128	86.3	85.2	85.5	85.3	11.0	256	

Table 3. Results of the sensitivity analysis for the YOWO model

Video clip condition		Training		Testing		Classification accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
		#clips	#frames	#clips	#frames				
Camera angle	High altitude	41	5,872	12	1,288	76.5	75.8	75.8	75.8
	Mid altitude	520	80,571	130	21,601	89.3	89.2	89.2	89.2
	Ground level	284	43,296	68	10,034	76.5	75.8	75.8	75.8
	Below ground level	4	490	1	143	82.6	82.5	82.5	82.5
Illumination	Low	54	7,160	20	2,471	78.3	77.7	77.7	77.7
	Mid	562	91,784	133	22,469	93.6	93.6	93.6	93.6
	High	233	31,285	58	8,126	83.3	83.0	83.0	83.0
	Contre-jour	15	2,399	5	691	92.9	92.9	92.9	92.0
Occlusion	Low	221	31,875	62	10,445	87.6	87.6	87.6	87.6
	Mid	58	8,995	13	1,891	75.8	75.8	75.8	75.8
	High	11	1,493	2	455	66.4	66.4	66.4	66.4
Weather condition	Cloudy	117	12,870	35	4,316	80.9	80.3	80.3	80.3
	Sunny	714	114,539	170	27,279	90.2	90.1	90.1	90.1
	Snow field	3	421	1	780	94.7	94.7	94.7	94.7
Resolution	Low	96	13,056	22	3,226	96.4	96.4	96.4	96.4
	High	753	117,173	189	29,840	88.2	87.9	87.9	87.9

6 Conclusions and future work

This paper investigates the benefits of using YOWO and YOWO53 methods over the state-of-the-art three-stage method for the activity recognition of excavators. The performance is evaluated using a custom dataset of 1060 videos collected from local construction sites and YouTube videos. The obtained results show that the joint optimization of single-stage methods (i.e., YOWO, YOWO53), provides significant performance improvement over the three-stage method, in which each stage is optimized separately. In particular, the YOWO model achieved an activity classification accuracy of 88.9% and an F1-score of 88.7%. In comparison, the YOWO53 model recorded slightly lower metrics, with both activity classification accuracy and F1-score at 85.4%. However, the best performance of the three-stage method was the activity classification accuracy, and F1-score of 70.4% and 69.8%, respectively.

Although both single-stage methods proved to be superior to the three-stage method, however, in contrast to the results obtained in a previous study for workers [16], the performance of the YOWO53 method was lower than that of the YOWO method when increasing the input size. Considering that the YOWO53 method was developed to improve the detection performance for small objects (i.e., workers), the performance gain over the YOWO method is only for the cases where either the object of interest or the input size is small, while the opposite behavior is seen for excavators. Hence, indicating to a possible shortcoming of the current single-stage methods and a possible future approach which is able to simultaneously recognize the activities of equipment and workers at different scales, especially for the interactive activities, which involve both workers and equipment. Therefore, a single network should be able to recognize both small (e.g., workers) as well as large (e.g., excavators) objects.

References

[1] Roberts D. and Golparvar-Fard M., "End-to-end vision-based detection, tracking and activity

analysis of earthmoving equipment filmed at ground level," *Automation in Construction*, vol. 105, p. 102811, Sep. 2019, doi: 10.1016/j.autcon.2019.04.006.

[2] Chen C., Zhu Z., and Hammad A., "Automated excavators activity recognition and productivity analysis from construction site surveillance videos," *Automation in Construction*, vol. 110, p. 103045, Feb. 2020, doi: 10.1016/j.autcon.2019.103045.

[3] Rezazadeh Azar E. and McCabe B., "Part based model and spatial–temporal reasoning to recognize hydraulic excavators in construction images and videos," *Automation in Construction*, vol. 24, pp. 194–202, Jul. 2012, doi: 10.1016/j.autcon.2012.03.003.

[4] Bohn J. S. and Teizer J., "Benefits and Barriers of Construction Project Monitoring Using High-Resolution Automated Cameras," *Journal of Construction Engineering and Management*, vol. 136, no. 6, pp. 632–640, Jun. 2010, doi: 10.1061/(ASCE)CO.1943-7862.0000164.

[5] Kim J., Chi S., and Seo J., "Interaction analysis for vision-based activity identification of earthmoving excavators and dump trucks," *Automation in Construction*, vol. 87, pp. 297–308, Mar. 2018, doi: 10.1016/j.autcon.2017.12.016.

[6] Kim J. and Chi S., "Action recognition of earthmoving excavators based on sequential pattern analysis of visual features and operation cycles," *Automation in Construction*, vol. 104, pp. 255–264, Aug. 2019, doi: 10.1016/j.autcon.2019.03.025.

[7] Chen C., Zhu Z., and Hammad A., "Critical Review and Road Map of Automated Methods for Earthmoving Equipment Productivity Monitoring," *Journal of Computing in Civil Engineering*, vol. 36, no. 3, p. 03122001, May 2022, doi: 10.1061/(ASCE)CP.1943-5487.0001017.

[8] Redmon J., Divvala S., Girshick R., and Farhadi A., "You Only Look Once: Unified, Real-Time Object Detection," In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, Jun. 2016, doi: 10.1109/CVPR.2016.91.

[9] Donahue J., Hendricks L. A., Guadarrama S., Rohrbach M., Venugopalan S., Darrell T., *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2625–2634, Jun. 2015, doi: 10.1109/CVPR.2015.7298878.

[10] Luo X., Li H., Cao D., Dai F., Seo J., and Lee S., "Recognizing Diverse Construction Activities in Site Images via Relevance Networks of Construction-Related Objects Detected by Convolutional Neural Networks," *Journal of Computing in Civil Engineering*, vol. 32, no. 3, p. 04018012, May 2018, doi: 10.1061/(ASCE)CP.1943-5487.0000756.

[11] Kim J. and Chi S., "Multi-camera vision-based productivity monitoring of earthmoving operations," *Automation in Construction*, vol. 112, p. 103121, Apr. 2020, doi: 10.1016/j.autcon.2020.103121.

[12] Slaton T., Hernandez C., and Akhavian R., "Construction activity recognition with convolutional recurrent networks," *Automation in Construction*, vol. 113, p. 103138, May 2020, doi: 10.1016/j.autcon.2020.103138.

[13] Chen C., Zhu Z., Hammad A., and Akbarzadeh M., "Automatic Identification of Idling Reasons in Excavation Operations Based on Excavator–Truck Relationships," *Journal of Computing in Civil Engineering*, vol. 35, no. 5, p. 04021015, Sep. 2021, doi: 10.1061/(ASCE)CP.1943-5487.0000981.

[14] Luo X., Li H., Yu Y., Zhou C., and Cao D., "Combining deep features and activity context to improve recognition of activities of workers in groups," *Computer-Aided Civil and Infrastructure Engineering*, vol. 35, no. 9, pp. 965–978, 2020, doi: 10.1111/mice.12538.

[15] Jung S., Jeoung J., Kang H., and Hong T., "3D convolutional neural network-based one-stage model for real-time action detection in video of construction equipment," *Computer-Aided Civil and Infrastructure Engineering*, vol. 37, no. 1, pp. 126–142, Jan. 2022, doi: 10.1111/mice.12695.

[16] Torabi G., Hammad A., and Bouguila N., "Two-Dimensional and Three-Dimensional CNN-Based Simultaneous Detection and Activity Classification of Construction Workers," *Journal of Computing in Civil Engineering*, vol. 36, no. 4, p. 04022009, Jul. 2022, doi: 10.1061/(ASCE)CP.1943-5487.0001024.

[17] Köpüklü O., Wei X., and Rigoll G., "You Only Watch Once: A Unified CNN Architecture for Real-Time Spatiotemporal Action Localization." arXiv, Oct. 18, 2021. doi: 10.48550/arXiv.1911.06644.

[18] Köpüklü O., Kose N., Gunduz A., and Rigoll G., "Resource Efficient 3D Convolutional Neural Networks," In *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 1910–1919, Oct. 2019, doi: 10.1109/ICCVW.2019.00240.

[19] Redmon J. and Farhadi A., "YOLO9000: Better, Faster, Stronger," In *IEEE Conference on*

Computer Vision and Pattern Recognition (CVPR), pages 6517–6525, Jul. 2017, doi: 10.1109/CVPR.2017.690.

[20] Girshick R., “Fast R-CNN,” In *IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.

[21] Redmon J. and Farhadi A., “Yolov3: An incremental improvement,” *arXiv preprint arXiv:1804.02767*, 2018.

[22] Wojke N., Bewley A., and Paulus D., “Simple online and realtime tracking with a deep association metric,” In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649, Sep. 2017, doi: 10.1109/ICIP.2017.8296962.

[23] Hara K., Kataoka H., and Satoh Y., “Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?,” In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018.

[24] Soomro K., Zamir A. R., and Shah M., “UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild,” *arXiv:1212.0402*, Dec. 2012, http://arxiv.org/abs/1212.0402

[25] Gu C., Sun C., Ross D. A., Vondrick C., Pantofaru C., Li Y., *et al.*, “Ava: A video dataset of spatio-temporally localized atomic visual actions,” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056, 2018.

[26] Carreira J., Noland E., Banki-Horvath A., Hillier C., and Zisserman A., “A Short Note about Kinetics-600,” *arXiv:1808.01340*, Aug. 2018, http://arxiv.org/abs/1808.01340

[27] Lin T.-Y., Maire M., Belongie S., Bourdev L., Girshick R., Hays J., *et al.*, “Microsoft COCO: Common Objects in Context,” *arXiv:1405.0312*, Feb. 2015, http://arxiv.org/abs/1405.0312

[28] Kingma D. and Ba J., “Adam: A Method for Stochastic Optimization,” *ICLR*, 2014.