Automation in Construction 130 (2021) 103872

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Vision–based framework for automatic interpretation of construction workers' hand gestures

Xin Wang, Zhenhua Zhu *

Department of Civil and Environmental Engineering, University of Wisconsin-Madison, 1415 Engineering Drive, Madison, WI 53706, USA

ARTICLE INFO

Keywords:
Construction automation
Visual detection and tracking
Hand gesture recognition
Human-machine interaction

ABSTRACT

Construction robots have been recently developed to improve construction productivity and safety. One of the critical steps to make the robots work with human workers as teams is to provide a user-friendly interface to support their mutual interactions on construction sites. Compared with existing interfaces, hand gestures are easy to use, natural, and intuitive. This paper proposed a novel vision-based framework to capture and interpret the worker's hand gestures as a human-robot interface in construction. The framework consists of three components: worker detection and tracking, recognition queues formulation, and hand gesture recognition. Its effectiveness on the hand gesture recognition was tested with field experiments and achieved the overall precision and recall of 87.0% and 66.7%. Also, a laboratory study was conducted to illustrate the use of the framework to interact with a robotic dump truck. Future work will integrate the proposed framework into robotic construction machines.

1. Introduction

The construction industry is falling behind others in terms of making productivity gains, maintaining onsite safety, attracting new laborers [1–3], etc. With years of technical development, construction robots and/or autonomous machines have shown the potential to increase construction productivity as well as solve problems such as labor shortage and safety risks in the construction field [4,5]. They have technical features to enhance the quality and efficiency of the operations and moreover could potentially perform construction tasks in dangerous or challenging environments [6,7].

Advances in robotics could not fully replace onsite workers. Instead, they make it possible for workers and robots to work collaboratively as a team [8,9]. This collaboration allows workers to off-load repetitive and tedious tasks to robots [8] and focus on those that cannot be easily performed by the robots [10]. One of the critical steps to leverage human-robot work collaboration is to provide a user-friendly interface to support their interactions. However, the field of the human-machine interface is less analyzed when developing robot technologies in construction [4].

The overall objective of this paper is to investigate the feasibility of interacting with robotic machines using hand gestures. The hand gesture is a common and natural form of human-machine interaction [11,12]. It provides a standard mode for workers to interact with construction robot machines. On construction sites, it could aid workers to convey correct directions without the need for complicated devices in a noisy environment [13].

So far, there are many research methods proposed for automatic hand gesture recognition using computer vision technologies. These methods either relied on hand-crafted visual features, such as improved dense trajectories (iDT) [14] and Mix Features Around Sparse Keypoints (MFSK) [15], or deep neural networks including 3D convolutional neural networks (CNNs) [11,16] and two-stream CNN architecture [17,18].

Existing methods for hand gesture recognition have been employed in various domains, including traffic police hand gesture identification, sign language understanding, human-machine interactions [19–21]. However, they are not appropriate to support human-robot collaboration in the construction fields due to the following reasons. First, existing methods tried to recognize a hand gesture through a single-time activation. Therefore, they cannot recognize the same gestures when they are consecutive. Second, these methods typically required a user to sit or stand still when making gestures. The recognition might fail when the user is moving. Moreover, their designs were to achieve high recognition accuracy. The balance between the accuracy and other factors, e.g. recognition speed and robustness was not well maintained.

* Corresponding author.
E-mail addresses: xwang2463@wisc.edu (X. Wang), zzhu286@wisc.edu (Z. Zhu).

https://doi.org/10.1016/j.autcon.2021.103872
Received 19 February 2021; Received in revised form 28 July 2021; Accepted 30 July 2021
Available online 6 August 2021

This paper proposed a vision-based framework for automatically capturing and interpreting the hand gestures of workers on construction sites to address the above-mentioned issues. The framework consisted of three components. First, the construction worker who gave hand gestures was visually detected and tracked in a camera video sequence to generate the regions of interest. Then, the regions were cropped to form hand gesture recognition queues based on the detection and tracking results. A hierarchical architecture was further constructed for the task of hand gesture detection and classification. The framework was tested with the videos collected from real construction sites and a laboratory experiment. The field test results showed the overall precision and recall achieved 87.0% and 66.7%, respectively. The laboratory experiment illustrated the potential of the framework to automate the hand gesture recognition for human-machine interaction in construction.

2. Related work

2.1. Visual detection of construction objects

The visual detection of construction objects (e.g. workers and equipment) is always a fundamental step in the process of automating construction engineering and management tasks in computer vision [22]. So far, there are many research studies that have been proposed to investigate the potential of visually detecting construction objects using background subtraction, through visual features, or by utilizing deep learning technologies. For background subtraction, the motion pixels in the video streams are firstly identified and extracted. Once the motion pixels are grouped, the regions of moving construction objects could be determined as foreground while other static regions are identified as background. For example, Chi and Caldas [23] presented an exploratory method of background subtraction to detect moving construction workers and equipment. The exploratory method distinguished foreground from background by using color information of image pixels. Another example can be found in the work of Gong and Caldas [24]. They evaluated and compared three kinds of background subtraction algorithms (e.g. Mixtures of Gaussian, Codebook based, and Bayesian model-based methods) on the task of detecting construction workers and equipment.

Besides, many studies which utilize visual features (e.g. shape and color) to detect construction objects have been performed and tested in the construction field. For shape-based features, histogram of oriented gradients (HOG) and Haar-like features are two widely adopted features. Rezazadeh Azar and McCabe [25] presented and evaluated two detection methods (e.g. HOG and Blob-HOG) for their ability to recognize dump trucks on construction sites. Park and Brilakis [26] trained Haar-like features with an Adaptive Boosting algorithm to detect and extract the shape information of construction equipment. Compared with shape features, the detection methods based on color features are generally more simple and effective, especially when the construction objects to detect are uniquely colored [22]. For example, Zou and Kim [27] employed HSV color space to extract the excavator of interest in the image data of foundation excavation activity. However, the methods which solely rely on color information might fail when construction activities could be conducted from day to night [22]. The detection performance could be further improved by combing the shape and color features as detection cues. An example can be found in the work of Memarzadeh et al. [28]. They proposed a novel detection method which relied on HOG and Colors (HOG +C) to detect construction workers and equipment.

Recently, deep learning technologies have been widely adopted in the visual detection of construction objects. For example, Luo et al. [29] employed Faster Region-based CNN (Faster R-CNN) to detect 22 categories of construction-related objects, such as equipment, materials and workers. Kim et al. [30] presented a CNN-based architecture to detect the excavator and truck on a construction site. The presented architecture consisted of the following three components: CNN methods for feature extraction, region proposal network for extracting candidate regions and constructing new feature maps, position-sensitive score maps and region of interest pooling layers for determining the object class. Son et al. [31] proposed a two-stage architecture for construction worker detection. In their work, feature maps and region proposals were firstly extracted from the scaled image via ResNet-152. Then, bounding box regression and region labeling were conducted with Faster R-CNN. Wu et al. [32] developed a CNN-based detection framework to automatically check whether individuals on construction sites were wearing hard hats. Different CNN layers were adopted to extract the features, which were then fused discriminately to generate a new feature pyramid. The fused feature pyramid was finally fed into the Single Shot Multibox Detector (SSD) to predict the final detection results.

2.2. Visual tracking of construction objects

Visual detection is mainly focused on determining and locating the objects in videos, while the visual tracking is generally utilized to identify and follow the movement trajectory of the objects [22,33]. Numerous studies about the tracking of construction objects have been performed in construction scenarios to measure productivity, assess material distribution, and monitor the site safety for engineers and managers. For example, Zou and Kim [27] adopted the Hue, Saturation, and Value (HSV) color space to track a hydraulic excavator and estimate its idling time. Park et al. [34] presented a comparative study of various vision tracker categories including contour-based, kernel-based and point-based methods to identify the most effective one in tracking construction resources.

Moreover, the tracking performance could be improved by the integration with the detection process. This idea has been tested in construction scenarios. Rezazadeh Azar et al. [35] proposed an automated tracking framework called server-customer interaction tracker (SCIT) to track dump trucks and measure the dirt loading cycles. The proposed framework combined the mean-shift tracking method with the HOG detection algorithm. Park and Brilakis [26] presented a novel hybrid method for tracking construction equipment that fused the detection and tracking algorithms. The detection algorithm located construction equipment by taking advantage of entities' motion, shape, and color distribution while the tracking algorithm stepped in the process to make up for the false detections.

In recent years, deep learning technologies have been employed to combine the tracking and detection process of construction objects. For example, Kim and Chi [36] used a Faster R-CNN detector and detection-based tracker to get the locations of the excavators. Firstly, excavators were detected in still images using a pre-trained Faster R-CNN detection model. The detection results were then associated to obtain the trajectories of the excavators using the Tracking-Learning-Detection algorithm. Luo et al. [37] employed the tracking-by-detection framework to implement multiple worker tracking. The framework detected workers in each frame with a deep learning technique, YOLOv3, and then tracked them across consecutive frames with a multiple object tracking method, SORT as the object estimation model. Angah and Chen [38] determined and followed the trajectories of multiple construction workers through deep learning and the gradient-based method. In their work, Mask R-CNN was utilized as the detector while the gradient-based method was employed to track workers by comparing the features of the detection results.

2.3. Vision-based hand gesture recognition

Hand gesture recognition is a hot topic in computer vision and pattern recognition, which plays an increasingly important role in the natural human-computer interface [39,40]. Currently, many efforts have been dedicated to vision-based hand gesture recognition. Using hand-crafted features and deep learning are two kinds of vision-based methods for hand gesture recognition. Traditional methods are

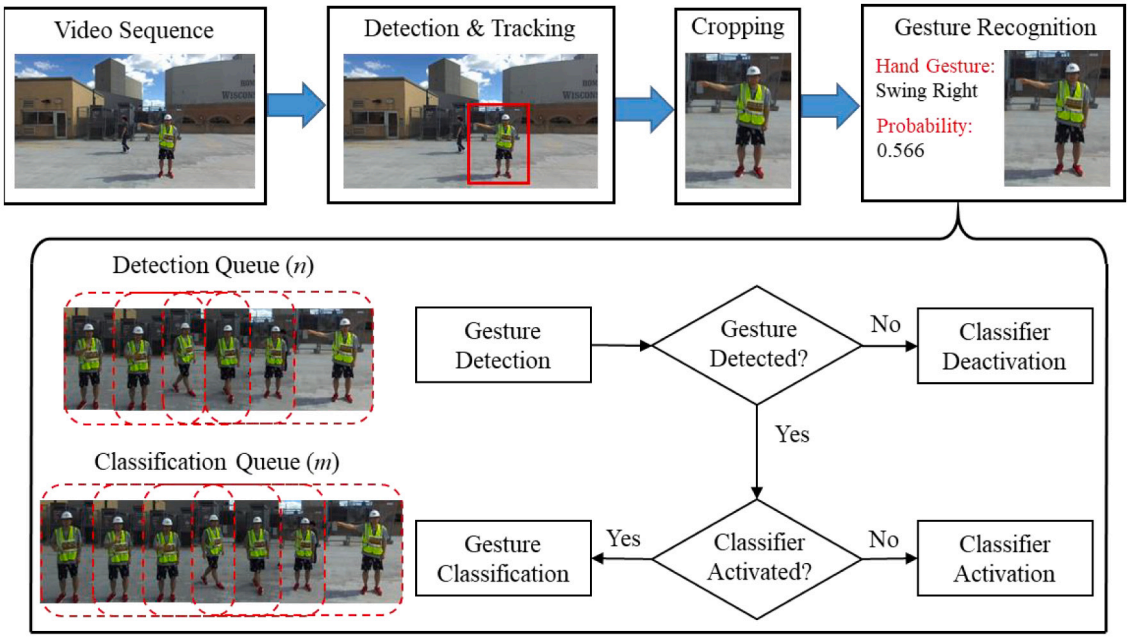


Fig. 1. The overview of the proposed framework.

generally based on hand-crafted features, such as HOG [41], iDT [14] and MFSK [15]. Besides, more studies were proposed to derive new sophisticated features which could represent the appearance, shape and/or motion changes of a gesture. For instance, Almeida et al. [42] presented a methodology for extracting hand gesture recognition features, such as hand area and hand movement velocity. These features were then fed into a SVM classifier to understand Brazilian sign language. Ahmed et al. [43] developed an integrated statistical algorithm which consisted of three modules: real-time detection of hand regions, hand trajectory tracking, and gesture recognition through the analysis of hand location variations. Memo and Zanuttigh [44] relied on the local curvature of a hand contour as feature descriptors and input them into an SVM classifier to achieve reliable, real-time hand gesture recognition.

So far, deep learning technologies have been widely used in hand gesture recognition. Typically, they can be achieved through 3D-CNN-based methods or two-stream CNN architecture. For 3D-CNN-based methods, one example can be found in the work of Miao et al. [45]. They firstly extracted spatiotemporal features using the Res3D network and combined the extracted features through a canonical correlation analysis. The final recognition was made by a linear SVM classifier. Similarly, Liao et al. [46] relied on the combination of a deep residual 3D-ConvNet and a bi-directional LSTM network to extract the spatio-temporal features of hand gestures from video sequences and score them accordingly for the automatic recognition of the sign language. Wang and Zhu [47] investigated and compared the performances of two 3D-CNN-based methods including ResNeXt-101 and Res3D + ConvLSTM+MobileNet on hand gesture recognition. In addition, a two-stream CNN architecture where two CNNs are adopted to model spatial and temporal information of sequences, separately, provides another technique for hand gesture recognition. For example, Huang et al. [18] developed a C3D network into a two-stream 3D-CNN architecture where one stream focused on the local, detailed hand gestures while the other stream was designed to extract global hand motions.

Several public datasets were established to measure the performance of the recognition methods. They could be classified into two categories including first-person view and second-person view. In the first-person view, a camera is typically mounted on the forehead of a user to imitate a viewpoint seen through the user's eyes. Examples of the first-person view dataset include the Interactive Museum dataset [48] and EgoGesture dataset [49]. Most of the datasets were captured in a second-person view, where a camera is kept a short distance towards a user. The user performs gestures actively like interacting with the camera. Several of them were created for general symbolic hand gestures, such as Cambridge Hand Gesture Dataset [50] and Sheffield Kinect Gesture (SKIG) Dataset [51]. Others were designed to support human-computer interaction including NvGesture Dataset [12] and IPN Hand Dataset [16], sign language interpretation like MSRGesture3D Dataset [52] and ChaLearn MMGR Dataset [53], etc.

2.4. Gaps in body of knowledge

The recent advance in computer vision technologies has built a solid foundation to capture and interpret the hand gestures of onsite construction workers and support their interactions with robotic machines. However, several challenges need to be addressed before making such interactions work well on construction sites. First, it is necessary to capture and interpret the hand gestures of workers when they are walking or under different postures. So far, existing hand gesture recognition methods were tested with the video clips of hand gestures that were typically recorded when subjects were sitting or standing still [12,16,49]. Their recognition performance was limited when workers are not sitting or standing still. Second, existing hand gesture methods were typically designed to recognize a gesture with one single-time activation [11,49]. As a result, they cannot recognize consecutive gestures made by workers. Moreover, most of the previous research focused on increasing the gesture classification accuracy by combining deep CNNs, which was forcing the limits of memory and increasing the reaction time [21,47]. To achieve the real-time response in practice, the recognition methods are supposed to strike a balance between acceptable classification accuracy and fast reaction.

3. Research objective and scope

The main objective of this study is to propose a novel vision-based framework that supports human-machine interaction on construction sites. The framework is expected to address the challenges mentioned above. Specifically, it will build a hand gesture recognition queue by combining the visual detection and tracking of the worker of interest. This way, the hand gestures of the worker under different postures could be captured. Also, it will redesign the hand gesture detection and

(a) Before horizontal extension (b) After horizontal extension

Fig. 2. The region of construction worker before and after the horizontal extension.

classification scheme to achieve a balance between acceptable classification accuracy and fast reaction. The framework will function in real construction scenarios, where workers are moving and making consecutive gestures simultaneously.

The focus of this study is placed on capturing and interpreting the hand gestures of construction workers for directing tower crane operations. These gestures are selected here since they are commonly seen on construction sites [47,54]. The framework could be expanded to capture and interpret other types of hand gestures without the loss of generality. In addition, it assumes that only video data are available as recognition cues. Workers are not requested to wear any wearable sensors for gesture recognition.

4. Proposed framework

The overview of the proposed framework is illustrated in Fig. 1. The framework consists of three components: visual detection and tracking of workers, frame cropping for recognition queue, and hand gesture recognition. Specifically, the construction worker who gives hand gestures is visually detected and tracked in a camera video sequence to generate the regions of interest. Based on the detection and tracking results, the regions are then cropped to form hand gesture recognition queues. Finally, a hierarchical architecture, which consists of a detector and a classifier, is employed to conduct the task of hand gesture detection and classification. More details are discussed in the following sections.

4.1. Visual detection and tracking of construction worker

The purpose of this component is to extract the construction worker who gives hand gestures in video sequences. A tracking-by-detection paradigm proposed in [55] is employed here due to its superior performance in tracking objects through long periods of occlusions. Within this paradigm, the detection module identifies the construction worker in each frame and obtains his/her bounding box. Given detection results, both trajectory and appearance information is modeled to associate current detections with existing tracks for the lifespan tracking of the worker. When there are multiple workers appearing in the scan, the construction worker who gives hand gestures could be identified through the tracking identification number (ID).

YOLOv3 [56] is selected to detect the construction worker because of its fast and accurate nature and ability to provide a multi-scale prediction. Additionally, many research results have verified the high performance of YOLOv3 in various construction object detections [37,57]. Multi-object deep Simple Online and Real-Time (SORT) tracker is employed to relate the same construction worker detected in the previous process across all the frames [55]. The deep SORT tracker is selected here since it is able to track the objects through long periods of occlusions and reduce the number of identity switches. Both trajectory and appearance information provided by the detection results are adopted to track the construction worker in video frames. More details about construction worker detection and tracking could be found in the work of [31,37,38].

4.2. Frame cropping for recognition queue

The purpose of this component is to crop the region of the construction worker who gives hand gestures from the original frame to form the queues for detecting and classifying hand gestures. This component can be divided into two steps: the horizontal extension of the extracted region, and the formation of the hand gesture recognition queues. The extracted region is firstly expanded horizontally by 25% to fully capture the hand gestures made by the worker based on trials and errors. As shown in Fig. 2 (a), the region which was directly obtained by the detection and tracking component might miss a part of the hand area when the worker was swinging his/her arms. After the horizontal extension both to the left and right (Fig. 2 (b)), the region of the construction worker could capture the whole hand area, which is crucial for the recognition of hand gestures.

Further, the cropped frames are compiled to form the hand gesture detection and classification queues. Both detection and classification queues take the current frame as a basis. The detection queue includes n previous frames, while the classification queue consists of m previous

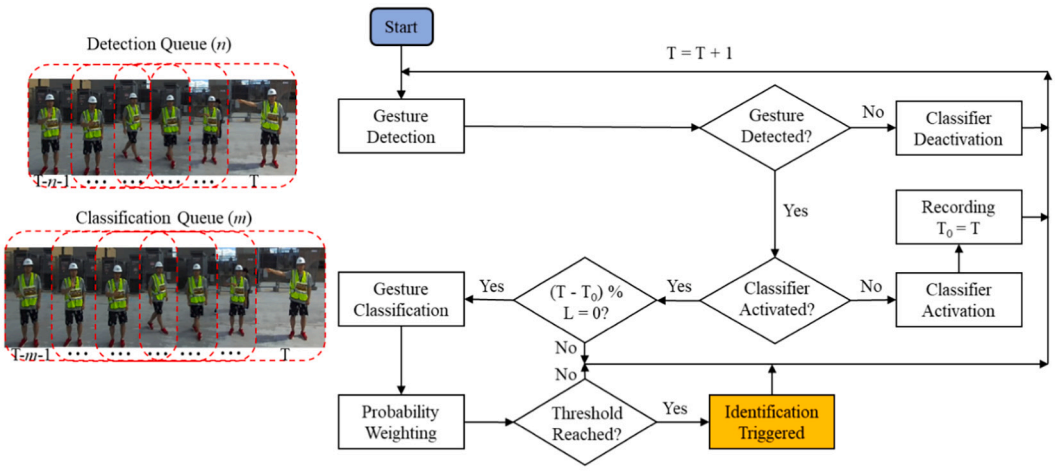


Fig. 3. The hierarchical architecture of hand gesture recognition.

Table 1
The networks of ResNet-10 and ResNeXt-101 (N_1, N_2, and F correspond to the number of ResNet blocks, ResNeXt blocks and feature channels, separately).

Layer Name	Conv1	Conv2_x	Conv3_x	Conv4_x	Conv5_x	–	Parameters
Output size	112×112	56×56	28×28	14×14	7×7	1×1	–
ResNet-10	Conv(3×7×7), Stride(1,2,2)	N_1: 1, F: 16	N_1: 1, F: 32	N_1: 1, F: 64	N_1: 1, F: 128	Average pooling, fc layer with softmax	862 K
ResNeXt-101		N_2: 3, N_2: 128	N_2: 24, F: 256	N_2: 36, F: 512	N_2: 3, F: 1024		47,497 K

frames. Following the guideline of [11], n is chosen as 8 frames since a smaller window size allows the detector to discover the start and end of the gestures more robustly. Besides, m is determined as 32 frames because the classification queue with 32 frames input achieves the best performance in [11]. The frames in the queues are further proportionally resized at a resolution of 112 × 112 pixels. After the formation of the recognition queues, they are input into the hand gesture recognition component.

4.3. Hand gesture recognition based on a hierarchical architecture

The purpose of this component is to employ a hierarchical CNN architecture to detect and classify the hand gestures made by the worker. By introducing a time factor L, the hierarchical architecture would function in identifying a consecutive gesture made by the worker. Fig. 3 shows the hierarchical architecture of hand gesture recognition, which consists of a detector and a classifier. The function of the detector is to detect whether there is a hand gesture in the detection queue. It is expected to be lightweight and run fast. This way, the video frames not containing any hand gesture could be discarded without being further processed. Here, the ResNet-10 model [11] is adopted since it requires small-sized features in each network layer.

When a hand gesture is detected, the classifier is utilized to classify its gesture class. The selection of the classifier should consider the balance between acceptable classification accuracy and fast reaction. Based on the findings from previous studies [11,16,47], ResNeXt-101 is selected. Table 1 summarized the specific network configurations of both ResNet-10 and ResNeXt-101.

The workflow of the hand gesture recognizing is illustrated in Fig. 3. It combines the gesture detector and classifier. The detector acts as a switch to decide whether the classifier needs to be activated. If a gesture gets detected and the classier has not been activated yet, the classifier will be activated and record the current frame index T as T_0. It refers to the first frame index when a gesture gets detected. Then, for the subsequent video frames received later, the classification queue will be input into the classifier to calculate the raw probability of each type of the hand gesture, only if the detector keeps detecting a gesture and the difference between the current frame index T and T_0 equals to a multiple of the time factor L. A weight function (Eq. 1) [11] is further applied to the raw classification probabilities to remove potential data noise.

$$w_T = \frac{1}{1 + e^{-0.2\times(T-T_0-u/(4\times s))}} \tag{1}$$

where w_T refers to the weight at frame T, u corresponds to the mean duration of the gestures (i.e. the number of frames) in the dataset, and s is the stride length which can be determined as 1 to be small enough not to miss any gestures [11].

The difference between the highest and the second-highest weighted probabilities is calculated. If this difference is more than a threshold τ, the identification of the hand gesture will be triggered; otherwise, it means that the classifier is not confident enough in classifying the hand gesture type. The architecture will conduct another round for the gesture detection and classification until the detector no longer detects the gesture and deactivates the classifier. It should be noted that the selection of τ and L depends on how likely and frequently the user intends to trigger the identification. Here, τ and L are chosen as 0.20 and 15 after trial and error.

Table 2
The parameter settings of the detector and classifier.

Components	Networks	Learning rate	Step size of learning rate decay	Batch size	Length of input frames
Detector	ResNet-10	0.01	10	8	8
Classifier	ResNeXt-101	0.01	15	20	32

5. Implementation and results

5.1. Implementation

The framework was implemented on an Ubuntu Linux 64-bit operating system. The Python 3.7 environment with the support of the Pytorch [58] and Tensorflow [59] platforms provides the critical algorithms, functions, and tools required for the framework. The hardware configuration has been listed as follows: an Intel® Core™ i7-4820K CPU (Central Processing Unit) @ 3.70 GHz, a 32 GB memory, and an NVIDIA Titan Xp DDR5X @ 12.0 GB GPU (Graphics Processing Unit).

5.2. Offline training for hand gesture recognition

The dataset created in [47] was employed to conduct the offline training for hand gesture recognition. The dataset contains hand gestures commonly made by workers on construction sites. In the dataset, the dynamic gestures were collected from 7 diverse indoor and outdoor scenes. There are a total of 364 video clips which include 1820 non-gesture samples and 1820 gesture samples. More details of the dataset could be found in the work of [47].

In order to train and test the detector, the gesture and non-gesture samples in the dataset were randomly split into the training subset (60%), validation subset (20%) and testing subset (20%). As for the training and testing of the classifier, only the gesture samples in the dataset were randomly divided into the training subset (60%), validation subset (20%) and testing subset (20%). Also, these samples were manually labeled based on their gesture types.

Table 2 summarized the parameters set for the training of the detector and classifier. The network configuration of the classifier is much more complicated than the detector, which typically requires more training data to prevent underfitting. Here, the transfer learning strategy was adopted for the classifier. The classifier is pre-trained firstly using the Jester dataset [60], which is the largest hand gesture dataset publicly available. Then, the specific training process for the detector and classifier is conducted as follows. The learning rate and the batch size are initially set as large as possible. The cross-entropy loss (Eq. 2) is employed as the loss function.

$$L_{CE} = -\sum_{i=1}^{n} t_i log(p_i) \tag{2}$$

where n is the number of classes, t_i is the truth label and p_i is the Softmax probability for the i-th class. When the training loss is steady, the learning rate is reduced with a fixed decay factor. Stochastic gradient descent (SGD) with Nesterov momentum of 0.9, damping factor of 0.9, and weight decay of 0.001 is employed as the optimizer. Moreover, all

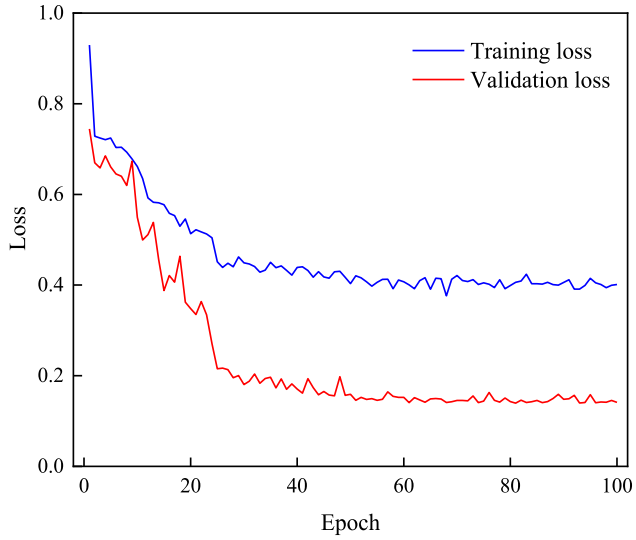


Fig. 4. The loss reduction along with the training progress for the detector.

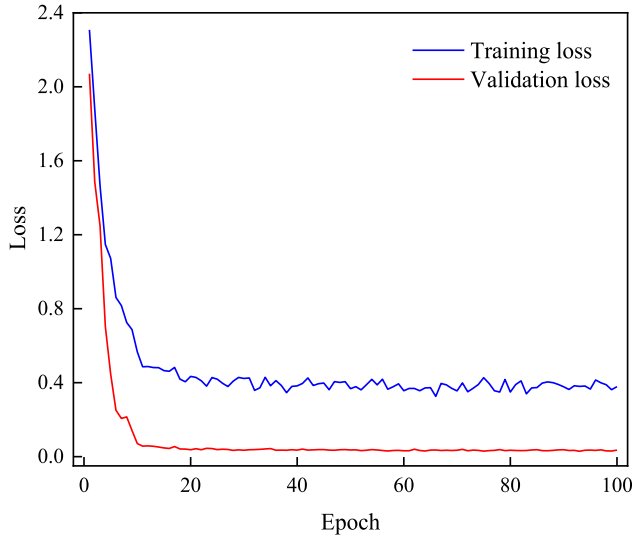
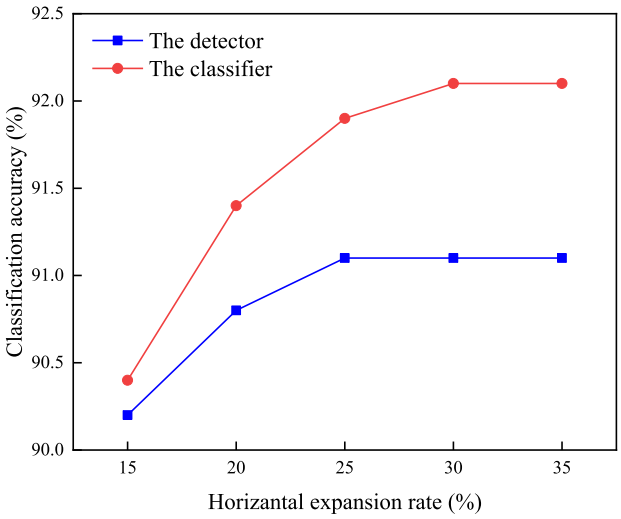


Fig. 5. The loss reduction along with the training progress for the classifier.

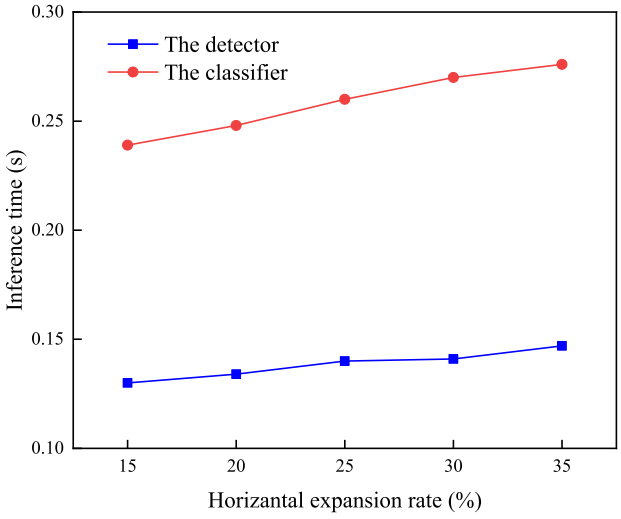
images of hand gesture samples are randomly cropped with a spatial size of 112 × 112 as the inputs for the data augmentation purpose. Figs. 4 and 5 show the loss reduction along with the training progress for the detector and classifier, respectively, when the horizontal expansion rate is 25%.

The training for the detectors and classifiers is conducted multiple times to investigate the effect of the horizontal expansion rates. The horizontal expansion rates are set as 15%, 20%, 25%, 30% and 35%, separately. Fig. 6 shows the variation of the classification accuracy and inference time with the expansion rate. Here, the classification accuracy is defined as the percentage of correctly labeled samples by the network. Inference time refers to the processing time of using a trained network to make one prediction. As indicated in Fig. 6, the classification accuracy becomes stable when the expansion rate reaches 25%. In the meantime, the inference time keeps increasing with the rise of the expansion rate. Considering the balance between the classification accuracy and inference time, the horizontal expansion rate is determined as 25%.

Table 3 presented the offline recognition performance of the detector and classifier with an expansion rate of 25%. The detector achieves a classification accuracy of 91.1% and an inference time of 0.14 s. For the classifier, the classification accuracy is 91.9% and the inference time obtains 0.26 s. The classifier achieves a higher classification accuracy but requires more inference time.



(a) The classification accuracy



(b) The inference time

Fig. 6. The variation of the classification accuracy and inference time with the expansion rate.

Table 3
The offline recognition performance of the detector and classifier.

Components	Networks	Classification accuracy (%)	Inference time (s)
Detector	ResNet-10	91.1	0.14
Classifier	ResNeXt-101	91.9	0.26

5.3. *Field experiment*

The effectiveness of the framework was tested through field experiments. The focus of the field experiment was placed on testing whether the framework could detect and track workers and capture and interpret their hand gestures on construction sites. A construction site near

Fig. 7. Examples of test scenarios.

Fig. 8. Examples of the field test results for subject 1.

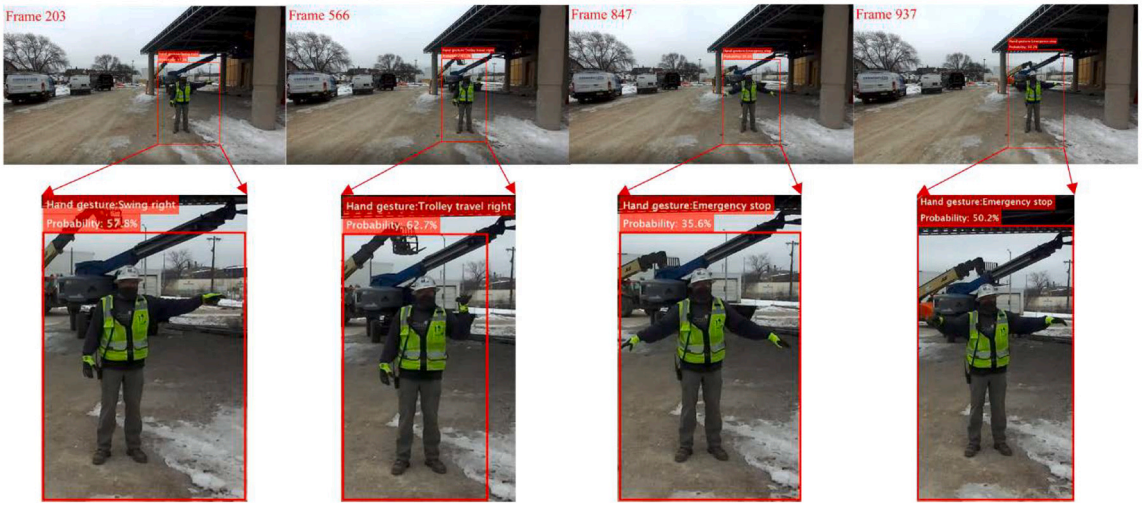


Fig. 9. Examples of the field test results for subject 2.

Milwaukee, WI. was selected for this field experiment. A ZED 2 stereo camera [61] was set up on the site to record the hand gestures made by construction workers. The maximum resolution of the videos could reach up to 2208 × 1242 pixels at 15 frames per second (fps). Six video clips were collected, which included 30 gesture samples in total. The examples of the test scenarios could be found in Fig. 7.

Figs. 8 and 9 showed two examples of testing the proposed framework to detect and track the workers (i.e. Subject 1 and 2) and then identify their hand gestures, e.g. "swing right" and "emergency stop". The corresponding test video clips could be found in the supplements. In each test, the detection and tracking results were represented within a series of bounding boxes along the video sequence. As shown in Fig. 10, when there are multiple workers in the scan, only the bounding box of the worker who makes hand gestures is returned based on his tracking ID. When the worker performed a hand gesture, the proposed framework triggered the gesture identification module and reported the corresponding gesture type.

Compared with the worker detection and tracking, the recognition of

Fig. 10. Identification of the signal man of interest.

the worker's hand gestures in construction has not been widely tested and evaluated before [47]. For this reason, the gesture identification component in the proposed framework was specifically evaluated here. Two quantitative indicators, i.e. precision and recall, were adopted and their definitions were given in Eqs. (3) and (4) [62].

$$\text{Precision}_i = \frac{TP_i}{TP_i + FP_i} \tag{3}$$

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i} \tag{4}$$

where i is the gesture class, TP_i is the number of gesture samples in gesture class i which are correctly predicted as i, FP_i is the number of non-gesture and gesture samples which are falsely predicted as i, and FN_i is the number of gesture samples in gesture class i which are falsely predicted as non-gesture or other gesture classes.

Table 4 compiled the recognition performance of the hand gesture under each type using the videos collected from the site. It was found that the overall precision and recall achieved 87.0% and 66.7%, respectively. The identification of "lower", "tower travel" and "dog everything" could reach up to 100% precision and 100% recall. The lowest precision and recall happened on the identification of "trolley travel right" and "swing left".

5.4. Pilot study

Further, a pilot study was conducted in a laboratory environment to test whether the framework proposed in this paper could serve as an interface to help workers control and/or interact with construction machines. In the study, the video frames were captured by the camera in real time. Each captured frame will be input into the framework immediately. Fig. 11 illustrated the setup of the laboratory experiment and the related data flow. Specifically, a subject was asked to perform hand gestures, which were captured by a video camera connecting to a computer. The captured hand gestures would be fed into the proposed framework and processed there in real time. Based on the recognition results, the corresponding instructions would be sent to a remote controller, where the control signals would be transmitted to operate the truck model remotely.

Fig. 12 showed an example of using the proposed framework to remotely control a toy truck to move and lift its dump box. The subject firstly made the hand gesture of "swing right" to request the truck model to turn right. The gesture was captured by the framework and the corresponding instruction was sent to the truck model through the remote

Table 4
The recognition performance of the performed hand gestures.

	Hoist	Lower	Tower travel	Trolley travel right	Trolley travel left	Dog everything	Swing right	Swing left	Emergency stop	Overall
# of samples	3	2	2	3	3	2	7	4	4	30
# of frames	271	173	143	250	339	215	753	512	455	3111
TP	2	2	2	1	2	2	5	1	3	20
FP	0	0	0	1	0	0	0	1	1	3
FN	1	0	0	2	1	0	2	3	1	10
Precision (%)	100.0	100.0	100.0	50.0	100.0	100.0	100.0	50.0	75.0	87.0
Recall (%)	66.7	100.0	100.0	33.3	66.7	100.0	71.4	25.0	75.0	66.7

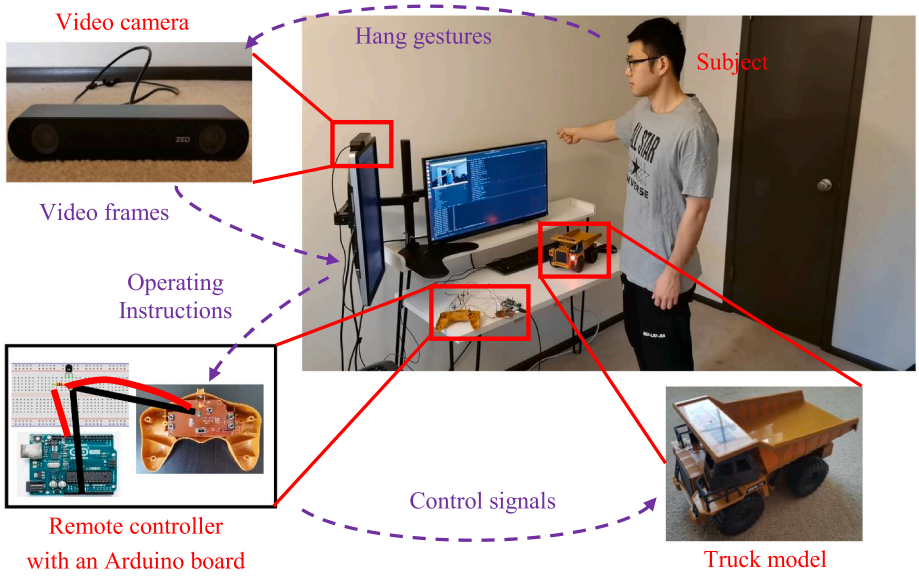


Fig. 11. Pilot study setup and data flow.

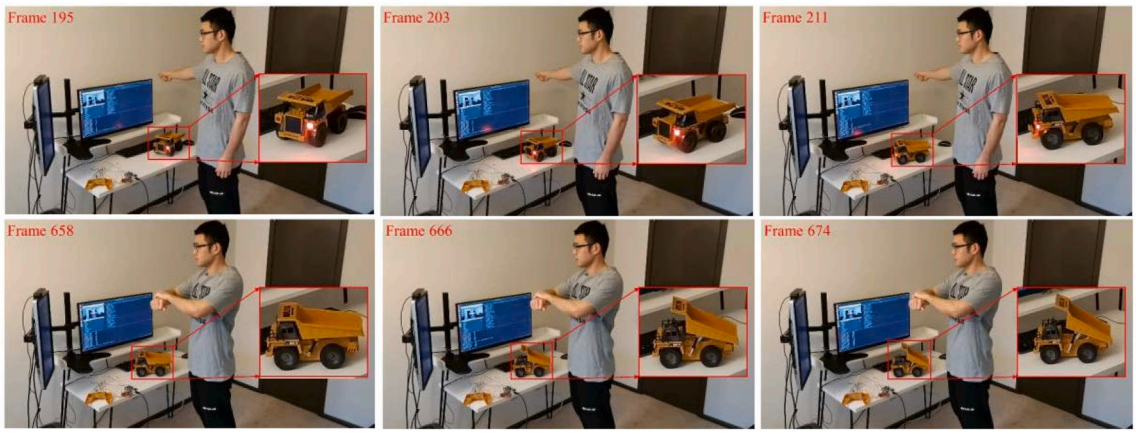


Fig. 12. Examples of the test results in the laboratory environment.

Fig. 13. Examples of false classifications.

Table 5
The performance comparison with different subjects in the field videos.

Indicators	Subject 1	Subject 2
Precision (%)	89.5	75.0
Recall (%)	68.0	60.0

controller in 1.5 ms. Following the instruction, the truck model drove towards the right gradually. After a short pause, the subject then performed the gesture of "dog everything" to request the truck model to lift its dump box. The truck model received the corresponding instruction in 1.4 ms and then lifted its dump box.

6. Discussion

Several lessons were learned from the field and laboratory experiments. First, hand gesture types impacted the recognition performance in the proposed framework. As illustrated in Table 4, the gesture types with relatively low recognition precisions are "trolley travel right" (50%), "swing left" (50%), and "Emergency stop" (75%). This is partly because the movements of these gestures are easily interfered by other gestures or body movements. For example, there was one false prediction of "swing left" as shown in Fig. 13 (left). The construction worker was managing his facial mask, which was similar to the beginning movement of the gesture "swing left". Another example could be found in Fig. 13 (right). The prediction of "emergency stop" was triggered by mistake while the subject was actually performing "swing left" since both gestures required the worker to raise and unroll his/her right arm.

Second, a diversity is expected when different subjects perform a same hand gesture. Therefore, the extensive training to capture this diversity plays an important role in improving the hand gesture recognition performance. Table 5 compared the recognition performance of the hand gestures made by two subjects. The recognition performance of

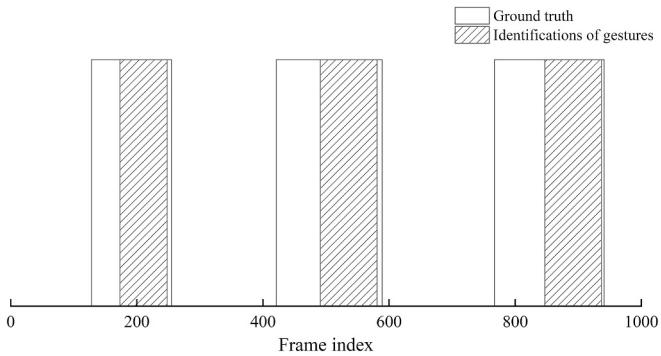


Fig. 14. The frame indices of the identifications.

the hand gestures made by subject 1 was superior to those made by subject 2. It may be because the hand gesture samples conducted by subject 2 were never used for the training of the hand gesture detector and classifier in the proposed framework. It is challenging for them to capture and differentiate the features of the gestures they had never seen before. The generalization issue is a universal problem for machine learning models. A wide range of classification models did not reach their original accuracy scores on unseen data. For example, it is reported that the accuracy drops of different models range from 3% to 15% on CIFAR-10 and 11% to 14% on ImageNet [63]. In this study, the drops for precision and recall are 14.5% and 8.0%, respectively, which are basically in an acceptable range.

Besides, the hand gesture was not recognized immediately after it was made. Here, the moment for triggering the hand gesture recognition was investigated. Fig. 14 indicated the frame indexes when hand gestures were started and finished for Subject 2. Compared with the ground truth, the recognition of a hand gesture was typically triggered only

when enough hand gesture motions were captured and interpreted. Typically, the recognition was always made 61 frames (approximately 4 s) later after the start of the hand gesture. The late recognition may be due to the preparation, nucleus and retraction parts from the beginning to the end of a dynamic gesture [64]. The nucleus is the most discriminative part while the other two parts can be quite similar for different gesture types. Thus, the classifier in the proposed framework can only make reliable classification decisions until the gesture enters its nucleus part. It should be noted that the late response is acceptable for most of the gesture types, such as "swing right" and "hoist", but may decrease the interaction efficiency when a gesture needs to be recognized as soon as possible like "emergency stop".

7. Conclusions and future work

The ideas of developing robotic machines have been recently proposed to promote automation and address issues, such as low productivity, poor safety records, labor shortage, etc. in the construction industries. One of the critical steps to make these machines work with onsite construction workers as teams is to provide a user-friendly interface to support their interactions. However, the field of the human-machine interface is less analyzed in the construction domain. This paper tried to fill the gap and proposed a vision-based framework to capture and interpret the hand gestures of construction workers to interact with construction robotic machines. The framework starts with the visual detection and tracking of a construction worker from a video sequence. Based on the detection and tracking results, the worker regions are cropped to form hand gesture recognition queues. Then, a hierarchical architecture is constructed to detect and classify the hand gestures made by the worker.

The framework has been tested with the videos collected from real construction sites and a laboratory experiment. The field test results showed the overall precision and recall achieved 87.0% and 66.7%, respectively. Also, the laboratory experiment indicated the framework could serve as an interface for the human-machine interaction. These results illustrated the potential of using computer vision technologies to automate the hand gesture recognition for human-machine interaction in construction. In addition, several lessons were learned from the field and laboratory experiments.

Future work will focus on two aspects. First, more construction scenarios and types of data sources will be included to make the training and testing of hand gesture detectors and classifiers more robust. The construction scenarios at different locations and with various weather conditions (e.g., rainy and snowy days) should be considered to enrich the dataset. Besides, the vision data could be further fused with wearable sensor data to help improve the generalization of gesture classifiers. Second, it will focus on developing a robotic system for workers to interact with real construction machines using the proposed framework. For example, the trajectory control techniques [65,66] could be integrated with the proposed framework and then employed in robot manipulators to simulate the operations of construction machines.

Supplementary data to this article can be found online at https://doi.org/10.1016/j.autcon.2021.103872.

Declaration of Competing Interest

None.

Acknowledgment

This paper is based in part upon the work supported by the Wisconsin Alumni Research Foundation (WARF) under Project No. AAD5524 and the M.A. Mortenson Company. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of WARF or Mortenson.

References

[1] P. Teicholz, Labor productivity declines in the construction industry: causes and remedies, in: AECbytes Viewpoint 67, 2004, p. 14. https://www.aecbytes.com/viewpoint/2013/issue_67.html. Accessed 20 Aug 2020.
[2] Y. Zhang, H. Zhu, Q. Guo, R. Carvel, Z. Yan, The effect of technical installations on evacuation performance in urban road tunnel fires, in: Tunneling and Underground Space Technology 107, 2021, p. 103608, https://doi.org/10.1016/j.tust.2020.103608.
[3] S. Kim, S. Chang, D. Castro-Lacouture, Dynamic modeling for analyzing impacts of skilled labor shortage on construction project management, J. Manag. Eng. 36 (2020), 04019035, https://doi.org/10.1061/(asce)me.1943-5479.0000720.
[4] J. Czarnowski, A. Dąbrowski, M. Maciaś, J. Główka, J. Wrona, Technology gaps in human-machine interfaces for autonomous construction robots, Autom. Constr. 94 (2018) 179–190, https://doi.org/10.1016/j.autcon.2018.06.014.
[5] B. Chu, K. Jung, M.T. Lim, D. Hong, Robot-based construction automation: an application to steel beam assembly (part I), Autom. Constr. 32 (2013) 46–61, https://doi.org/10.1016/j.autcon.2012.12.016.
[6] N. Melenbrink, J. Werfel, A. Menges, On-site autonomous construction robots: towards unsupervised building, Autom. Constr. 119 (2020) 103312, https://doi.org/10.1016/j.autcon.2020.103312.
[7] H. Ardiny, S. Witwicki, F. Mondada, Construction automation with autonomous mobile robots: a review, in: International Conference on Robotics and Mechatronics (ICROM), 2015, pp. 418–424, https://doi.org/10.1109/ICRoM.2015.7367821.
[8] S. You, J.H. Kim, S.H. Lee, V. Kamat, L.P. Robert, Enhancing perceived safety in human–robot collaborative construction using immersive virtual environments, Autom. Constr. 96 (2018) 161–170, https://doi.org/10.1016/j.autcon.2018.09.008.
[9] A. Bauer, D. Wollherr, M. Buss, Human-robot collaboration: a survey, Int. J. Humanoid Robot. 5 (2008) 47–66, https://doi.org/10.1142/S0219843608001303.
[10] S. You, T. Ye, L.P. Robert, Team potency and ethnic diversity in embodied physical action (EPA) robot-supported dyadic teams, in: ICIS 2017 Transforming Society with Digital Innovation, 2017. http://aisel.aisnet.org/icis2017/HumanBehavior/Presentations/3/. Accessed 15 Aug 2020.
[11] O. Köpüklü, A. Gunduz, N. Kose, G. Rigoll, Real-time hand gesture detection and classification using convolutional neural networks, in: 14th IEEE International Conference on Automatic Face and Gesture Recognition, 2019, pp. 1–8, https://doi.org/10.1109/FG.2019.8756576.
[12] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, J. Kautz, Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural networks, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 4207–4215, https://doi.org/10.1109/CVPR.2016.456.
[13] P.E. Hagan, J.F. Montgomery, J.T. O'Reilly, Accident Prevention Manual for Business & Industry: Engineering & Technology, National Safety Council, 2015. ISBN: 9780879123222.
[14] H. Wang, D. Oneata, J. Verbeek, C. Schmid, A robust and efficient video representation for action recognition, Int. J. Comput. Vis. 119 (2016) 219–238, https://doi.org/10.1007/s11263-015-0846-5.
[15] J. Wan, G. Guo, S.Z. Li, Explore efficient local features from RGB-D data for one-shot learning gesture recognition, IEEE Trans. Pattern Anal. Mach. Intell. 38 (2015) 1626–1639, https://doi.org/10.1109/TPAMI.2015.2513479.
[16] G. Benitez-Garcia, J. Olivares-Mercado, G. Sanchez-Perez, K. Yanai, IPN hand: a video dataset and benchmark for real-time continuous hand gesture recognition, in: 25th IEEE International Conference on Pattern Recognition (ICPR), 2020, pp. 4340–4347, https://doi.org/10.1109/ICPR48806.2021.9412317.
[17] J. Wu, P. Ishwar, J. Konrad, Two-stream CNNs for gesture-based verification and identification: learning user style, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2016, pp. 42–50, https://doi.org/10.1109/CVPRW.2016.21.
[18] J. Huang, W. Zhou, Q. Zhang, H. Li, W. Li, Video-based sign language recognition without temporal segmentation, in: 32nd AAAI Conference on Artificial Intelligence, 2018. https://ojs.aaai.org/index.php/AAAI/article/view/11903. Accessed 21 Jul 2020.
[19] P. Kumar, H. Gauba, P. Pratim Roy, D. Prosad Dogra, A multimodal framework for sensor based sign language recognition, Neurocomputing 259 (2017) 21–38, https://doi.org/10.1016/j.neucom.2016.08.132.
[20] F. Guo, J. Tang, X. Wang, Gesture recognition of traffic police based on static and dynamic descriptor fusion, Multimed. Tools Appl. 76 (2017) 8915–8936, https://doi.org/10.1007/s11042-016-3497-9.
[21] P. Narayana, J.R. Beveridge, B.A. Draper, Gesture recognition: focus on the hands, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5235–5244, https://doi.org/10.1109/CVPR.2018.00549.
[22] Z. Zhu, X. Ren, Z. Chen, Integrated detection and tracking of workforce and equipment from construction jobsite videos, Autom. Constr. 81 (2017) 161–171, https://doi.org/10.1016/j.autcon.2017.05.005.
[23] S. Chi, C.H. Caldas, Automated object identification using optical video cameras on construction sites, Comput. Aided Civil Infrastruct. Eng. 26 (2011) 368–380, https://doi.org/10.1111/j.1467-8667.2010.00690.x.
[24] J. Gong, C.H. Caldas, An object recognition, tracking, and contextual reasoning-based video interpretation method for rapid productivity analysis of construction operations, Autom. Constr. 20 (2011) 1211–1226, https://doi.org/10.1016/j.autcon.2011.05.005.
[25] E. Rezazadeh Azar, B. McCabe, Automated visual recognition of dump trucks in construction videos, J. Comput. Civ. Eng. 26 (2012) 769–781, https://doi.org/10.1061/(asce)cp.1943-5487.0000179.

[26] M.W. Park, I. Brilakis, Enhancement of construction equipment detection in video frames by combining with tracking, in: Congress of Computing in Civil Engineering, 2012, pp. 421–428, https://doi.org/10.1061/9780784412343.0053.
[27] J. Zou, H. Kim, Using hue, saturation, and value color space for hydraulic excavator idle time analysis, J. Comput. Civ. Eng. 21 (2007) 238–246, https://doi.org/10.1061/(asce)0887-3801(2007)21:4(238).
[28] M. Memarzadeh, A. Heydarian, M. Golparvar-Fard, J.C. Niebles, Real-time and automated recognition and 2D tracking of construction workers and equipment from site video streams, in: Congress of Computing in Civil Engineering, 2012, pp. 429–436, https://doi.org/10.1061/9780784412343.0054.
[29] X. Luo, H. Li, D. Cao, F. Dai, J. Seo, S. Lee, Recognizing diverse construction activities in site images via relevance networks of construction-related objects detected by convolutional neural networks, J. Comput. Civ. Eng. 32 (2018), 04018012, https://doi.org/10.1061/(asce)cp.1943-5487.0000756.
[30] H. Kim, S. Bang, H. Jeong, Y. Ham, H. Kim, Analyzing context and productivity of tunnel earthmoving processes using imaging and simulation, Autom. Constr. 92 (2018) 188–198, https://doi.org/10.1016/j.autcon.2018.04.002.
[31] H. Son, H. Choi, H. Seong, C. Kim, Detection of construction workers under varying poses and changing background in image sequences via very deep residual networks, Autom. Constr. 99 (2019) 27–38, https://doi.org/10.1016/j.autcon.2018.11.033.
[32] J. Wu, N. Cai, W. Chen, H. Wang, G. Wang, Automatic detection of hardhats worn by construction personnel: a deep learning approach and benchmark dataset, Autom. Constr. 106 (2019) 102894, https://doi.org/10.1016/j.autcon.2019.102894.
[33] J. Yang, M.W. Park, P.A. Vela, M. Golparvar-Fard, Construction performance monitoring via still images, time-lapse photos, and video streams: now, tomorrow, and the future, Adv. Eng. Inform. 29 (2015) 211–224, https://doi.org/10.1016/j.aei.2015.01.011.
[34] M.W. Park, A. Makhmalbaf, I. Brilakis, Comparative study of vision tracking methods for tracking of construction site resources, Autom. Constr. 20 (2011) 905–915, https://doi.org/10.1016/j.autcon.2011.03.007.
[35] E. Rezazadeh Azar, S. Dickinson, B. McCabe, Server-customer interaction tracker: computer vision–based system to estimate dirt-loading cycles, J. Constr. Eng. Manag. 139 (2013) 785–794, https://doi.org/10.1061/(asce)co.1943-7862.0000652.
[36] J. Kim, S. Chi, Action recognition of earthmoving excavators based on sequential pattern analysis of visual features and operation cycles, Autom. Constr. 104 (2019) 255–264, https://doi.org/10.1016/j.autcon.2019.03.025.
[37] X. Luo, H. Li, H. Wang, Z. Wu, F. Dai, D. Cao, Vision-based detection and visualization of dynamic workspaces, Autom. Constr. 104 (2019) 1–13, https://doi.org/10.1016/j.autcon.2019.04.001.
[38] O. Angah, A.Y. Chen, Tracking multiple construction workers through deep learning and the gradient based method with re-matching based on multi-object tracking accuracy, Autom. Constr. 119 (2020) 103308, https://doi.org/10.1016/j.autcon.2020.103308.
[39] S.E. Ovur, H. Su, W. Qi, E. De Momi, G. Ferrigno, Novel adaptive sensor fusion methodology for hand pose estimation with multileap motion, IEEE Trans. Instrum. Meas. 70 (2021) 1–8, https://doi.org/10.1109/TIM.2021.3063752.
[40] W. Qi, H. Su, A. Aliverti, A smartphone-based adaptive recognition and real-time monitoring system for human activities, in: IEEE Transactions on Human-Machine Systems 50, 2020, pp. 414–423, https://doi.org/10.1109/THMS.2020.2984181.
[41] E. Ohn-Bar, M.M. Trivedi, Hand gesture recognition in real time for automotive interfaces: a multimodal vision-based approach and evaluations, IEEE Trans. Intell. Transp. Syst. 15 (2014) 2368–2377, https://doi.org/10.1109/TITS.2014.2337331.
[42] S.G.M. Almeida, F.G. Guimarães, J. Arturo Ramírez, Feature extraction in Brazilian sign language recognition based on phonological structure and using RGB-D sensors, Expert Syst. Appl. 41 (2014) 7259–7271, https://doi.org/10.1016/j.eswa.2014.05.024.
[43] W. Ahmed, K. Chanda, S. Mitra, Vision based hand gesture recognition using dynamic time warping for Indian sign language, in: 2016 IEEE International Conference on Information Science (ICIS), 2016, pp. 120–125, https://doi.org/10.1109/INFOSCI.2016.7845312.
[44] A. Memo, P. Zanuttigh, Head-mounted gesture controlled interface for human-computer interaction, Multimed. Tools Appl. 77 (2018) 27–53, https://doi.org/10.1007/s11042-016-4223-3.
[45] Q. Miao, Y. Li, W. Ouyang, Z. Ma, X. Xu, W. Shi, X. Cao, Multimodal gesture recognition based on the ResC3D network, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2017, pp. 3047–3055, https://doi.org/10.1109/ICCVW.2017.360.
[46] Y. Liao, P. Xiong, W. Min, W. Min, J. Lu, Dynamic sign language recognition based on video sequence with BLSTM-3D residual networks, IEEE Access 7 (2019) 38044–38054, https://doi.org/10.1109/ACCESS.2019.2904749.
[47] X. Wang, Z. Zhu, Vision-based hand signal recognition in construction: a feasibility study, Autom. Constr. 125 (2021) 103625, https://doi.org/10.1016/j.autcon.2021.103625.
[48] L. Baraldi, F. Paci, G. Serra, L. Benini, R. Cucchiara, Gesture recognition in ego-centric videos using dense trajectories and hand segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 688–693, https://doi.org/10.1109/CVPRW.2014.107.
[49] Y. Zhang, C. Cao, J. Cheng, H. Lu, EgoGesture: a new dataset and benchmark for egocentric hand gesture recognition, in: IEEE Transactions on Multimedia 20, 2018, pp. 1038–1050, https://doi.org/10.1109/TMM.2018.2808769.
[50] T.K. Kim, R. Cipolla, Canonical correlation analysis of video volume tensors for action categorization and detection, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2009) 1415–1428, https://doi.org/10.1109/TPAMI.2008.167.
[51] L. Liu, L. Shao, Learning discriminative representations from RGB-D video data, in: Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, 2013, pp. 1493–1500, https://doi.org/10.5555/2540128.2540343. Accessed 26 Aug 2020.
[52] A. Kurakin, Z. Zhang, Z. Liu, A real time system for dynamic hand gesture recognition with a depth sensor, in: 2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO), 2012, pp. 1975–1979. https://ieeexplore.ieee.org/document/6333871. Accessed 6 Jun 2020.
[53] S. Escalera, X. Baró, J. Gonzàlez, M.A. Bautista, M. Madadi, M. Reyes, V. Ponce-López, H.J. Escalante, J. Shotton, I. Guyon, Chalearn looking at people challenge 2014: dataset and results, in: European Conference on Computer Vision, 2015, pp. 459–473, https://doi.org/10.1007/978-3-319-16178-5_32.
[54] The American Society of Mechanical Engineers, Safety Standard for Cableways, Cranes, Derricks, Hoists, Hooks, Jacks, and Slings, 2012. ISBN: 9780791829073.
[55] N. Wojke, A. Bewley, D. Paulus, Simple online and realtime tracking with a deep association metric, in: 2017 IEEE international conference on image processing (ICIP), 2018, pp. 3645–3649, https://doi.org/10.1109/ICIP.2017.8296962.
[56] J. Redmon, A. Farhadi, YOLOv3: an incremental improvement, ArXiv (2018), 1804.02767, https://arxiv.org/abs/1804.02767. Accessed 6 Jul 2020.
[57] F. Wu, G. Jin, M. Gao, Z. He, Y. Yang, Helmet detection based on improved YOLO V3 deep model, in: 2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC), 2019, pp. 363–368, https://doi.org/10.1109/ICNSC.2019.8743246.
[58] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: an imperative style, high-performance deep learning library, Advan. Neural Info. Process. Syst. 32 (2019) 8026–8037. https://arxiv.org/abs/1912.01703. Accessed 18 Jul 2020.
[59] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D.G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, X. Zheng, TensorFlow: a system for large-scale machine learning, in: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 2016, pp. 265–283, https://doi.org/10.5555/3026877.3026899. Accessed 11 May 2020.
[60] J. Materzynska, G. Berger, I. Bax, R. Memisevic, The jester dataset: a large-scale video dataset of human gestures, in: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2019, pp. 2874–2882, https://doi.org/10.1109/ICCVW.2019.00349.
[61] Stereolabs, ZED 2-AI Stereo Camera. https://www.stereolabs.com/zed-2/, 2019. Accessed 18 May 2020.
[62] J. Davis, M. Goadrich, The relationship between precision-recall and ROC curves, in: Proceedings of the 23rd International Conference on Machine learning, 2006, pp. 233–240, https://doi.org/10.1145/1143844.1143874.
[63] R. Roelofs, Measuring Generalization and Overfitting in Machine Learning, Doctoral Dissertation, UC Berkeley, 2019, https://escholarship.org/uc/item/6j01x9mz. Accessed 29 Dec 2020.
[64] V.I. Pavlovic, R. Sharma, T.S. Huang, Visual interpretation of hand gestures for human-computer interaction: a review, IEEE Trans. Pattern Anal. Mach. Intell. 19 (1997) 677–695, https://doi.org/10.1109/34.598226.
[65] H. Su, A. Mariani, S.E. Ovur, A. Menciassi, G. Ferrigno, E. De Momi, Toward teaching by demonstration for robot-assisted minimally invasive surgery, IEEE Trans. Autom. Sci. Eng. 18 (2021) 484–494, https://doi.org/10.1109/TASE.2020.3045655.
[66] H. Su, W. Qi, Y. Hu, H.R. Karimi, G. Ferrigno, E. De Momi, An incremental learning framework for human-like redundancy optimization of anthropomorphic manipulators, in: IEEE Transactions on Industrial Informatics, 2020, https://doi.org/10.1109/TII.2020.3036693.