Sensors and Materials, Vol. 36, No. 2 (2024) 729–743
MYU Tokyo

S & M 3557

A Smart Assembly Line Design Using Human–Robot Collaborations with Operator Gesture Recognition by Decision Fusion of Deep Learning Channels of Three Image Sensing Modalities from RGB-D Devices

Ing-Jr Ding[1*] and Ya-Cheng Juang[2]

[1]Department of Electronic Engineering, National United University,
No. 2, Lienda, Miaoli City, Taiwan
[2]Department of Electrical Engineering, National Formosa University,
No. 64, Wunhua Rd., Huwei Township, Yunlin County, Taiwan

(Received September 3, 2023; accepted February 19, 2024)

Keywords: assembly line, human–robot collaboration, RGB-D image sensor, operator gesture recognition, deep learning, decision fusion

Machine vision with image sensors has been employed in smart manufacturing such as the popular automatic optical inspection (AOI) by deploying an image acquisition camera to optically scan the target device for quality defects. With the rapid progress of image sensor techniques, the RGB-D image sensor device that can capture operator assembly gesture actions to make intelligent interactions between a robot and an operator has been developed. In this study, we propose a smart assembly-line design for intelligent manufacturing or factory applications where a working mode of human–robot collaboration (HRC) will be incorporated. In the proposed HRC assembly line, the operator and manipulator (robotic arm) will co-work with each other where the appropriately deployed RGB-D image device (the well-known Intel RealSense camera in this work) is used to acquire assembly gesture data of the operator to further perform operator gesture recognition. The manipulator will then perform the corresponding feedback action according to the recognized operation gesture (e.g., grabbing the scissors and then moving to the operator if the gesture of winding the tape is recognized). For operator gesture recognition, we first construct three different sensing modalities of deep learning recognition channels, which are the RGB convolution neural network (CNN)-long short-term memory (LSTM) channel with RGB gesture image inputs, the depth CNN-LSTM channel with depth gesture image inputs, and the 3D-(x, y, z) LSTM raw channel with skeleton raw data inputs. A decision fusion scheme is then developed for hybridizations of recognition decision outputs of these three separated deep learning gesture recognition channels with different gesture sensing modalities. In this work, various weight combination strategies to achieve the decision fusion of three deep learning recognition channels are used to evaluate the effectiveness of operator gesture recognition. Experiments on classifications of ten categories of operator assembly gestures show that the half-quarter-quarter strategy with the setting of

*Corresponding author: e-mail: eugen.ding@gmail.com
https://doi.org/10.18494/SAM4788

ISSN 0914-4935
https://myukk.org/

(w_{RGB}, w_{Depth}, w_{3D}) = (0.5, 0.25, 0.25) for weight allocations of channel decisions can achieve the highest recognition accuracy.

1. Introduction

As conventional factory operators perform repetitive tasks, the smart factory with automation is adapting to changing demands in real time. Within the smart factory, the robotic arm (also known as the manipulator), equipped with sensors and advanced algorithms, performs tasks that once relied heavily on human intervention. The automatic optical inspection (AOI) that uses specialized cameras (or image sensors) hybridized with image process algorithms to scan and analyze every intricate detail of the manufactured components to ensure flawless quality is a typical application and has been widely used in smart manufacturing. Human–robot collaboration (HRC) that belongs to an innovative mission completion mode is accelerating its introduction into the field of smart manufacturing. In the HRC with dynamic collaborations between the operator and the robot, the robot endowed with artificial intelligence deftly handles intricate tasks and assists the operator, while the human operator provides strategic oversight, creativity, and problem-solving expertise. To enhance the classical AOI, one of the authors has proposed in a previous work a human manipulator collaboration-based scheme for object inspection where hand pose sensing images of numeric symbols combined with non-numeric expressions made by the operator to indicate the type and quality of the inspected object are recognized, and the manipulator with the grabbed object will then move and release the object to the correct region according to hand pose recognition results.[1] Such an HRC scheme can be further extended and used in the assembly line of smart manufacturing to complete the assembly tasks of a specific device (e.g., to assemble a computer or a circuit board with electronic components) by the fine co-working of the operator and robot.

Hand gesture information from the operator usually reveals the practical intention of the operator, and therefore, such a sensing clue will play an important role in the design of HRC-based assembly-line mission completion systems. With a clear understanding of various specific operator hand gestures, the robot will also be able to participate in the operator assembly process and make a corresponding appropriate reaction in a suitable time. An example of an interaction between a human and a robot in the HRC-based task is that when the system realizes that the operator has made a gesture of winding a tape, the manipulator will grab a pair of scissors and then move to the operator immediately. In the studies described in Refs. 2–6, the HRC strategy is incorporated into the assembly-line mission. In the work described in Ref. 2, the hidden Markov model (HMM) is used for hand gesture modeling and recognition to construct the human–robot interactive assembly. The study described in Ref. 3 provides an integrated framework for human–robot collaborative assembly. The assembly workstation task by HRC is shown in Ref. 4. In Refs. 5 and 6, the authors established a human–robot collaborative assembly with gesture recognition by capturing wearable device and depth-camera gesture data, respectively. Related applications that use gesture recognition to implement the HRC are also described in Refs. 7–12 (on-line programming of industrial robot manipulators in Ref. 7, plan recognition and trajectory prediction in Ref. 8, an assistive scenario with the robot in Ref. 9,

sensing of double-hand gestures by a Kinect device to control a robotic arm in Ref. 10, solutions of multi-user online recognition of technical gestures for HRC in manufacturing in Ref. 11, and providing support to the operator by AI-enhanced wearable devices in Ref. 12). All these studies of HRC with gesture recognition for assembly-line or other related application scenarios use only conceptually simple approaches or the data statistics strategy (such as HMM) for establishing the gesture recognition system. Owing to the lack of detailed learning of the captured operator gesture data, such a constructed gesture recognition system will clearly be less competitive in terms of gesture recognition accuracy. Gesture recognition with substandard performance will considerably increase the operational difficulty and decrease the market acceptability, which is undoubtedly a restriction for HRC assembly-line systems to be practically applied to the real world. To tackle this issue of unreliable gesture recognition results, we developed a deep learning approach for performing accurate classifications of the assembly gesture actions of the operator in the HRC assembly line where three different image sensing modalities of operator gesture data, namely, RGB images, depth images, and 3D skeleton raw data, are acquired simultaneously by the RGB-D image sensor and sent to three well-trained separated deep learning channels of the RGB convolution neural network (CNN)-long short-term memory (LSTM), the depth CNN-LSTM, and the 3D-(x, y, z) raw LSTM, respectively, for gesture classifications. Aimed at the three modalities of decisions made from three corresponding deep learning recognition channels, we present various weight combination strategies to achieve channel decision fusion and then determine a much reliable recognition outcome. In fact, the use of deep learning techniques for building up gesture recognition systems has been explored in the authors' previous related studies.[13–16] Those works employ only a single modality of gesture data (the wearable-device-derived or image-sensor-derived data) or the fused feature of different sensing modalities to construct a unique deep learning recognition channel for classifying the input gesture. In the following sections, we will describe in detail such HRC assembly-line design with operator gesture recognition by the decision fusion of three separated deep learning recognition channels, namely, RGB CNN-LSTM, depth CNN-LSTM, and 3D-(x, y, z) raw LSTM.

2. HRC-based Strategies with Operator Gesture Recognition for Advanced Assembly Lines of Smart Manufacturing

Figure 1 shows the practical application scenario of the presented HRC-based assembly line with operator gesture recognition. The presented HRC-based assembly-line scheme in this study can allow the operator and manipulator to cooperatively complete the task. In this work, a computer assembly task is set as the mission to be completed. The manipulator used in this work is "OpenMANIPULATOR-X," which is made by the company Robotis. OpenMANIPULATOR-X is an open-source robotic arm based on a robot operating system (ROS) and has five active motors (5 DOF in total, 4 DOF + 1 DOF Gripper), each of which is the DYNAMIXEL XM-430 model.[17] Table 1 shows all significant gesture actions performed by the operator (ten assembly actions in total defined in this study, including acquiring the object released from the manipulator, winding the tape, pulling the cable tie, and picking up desired items that are put in

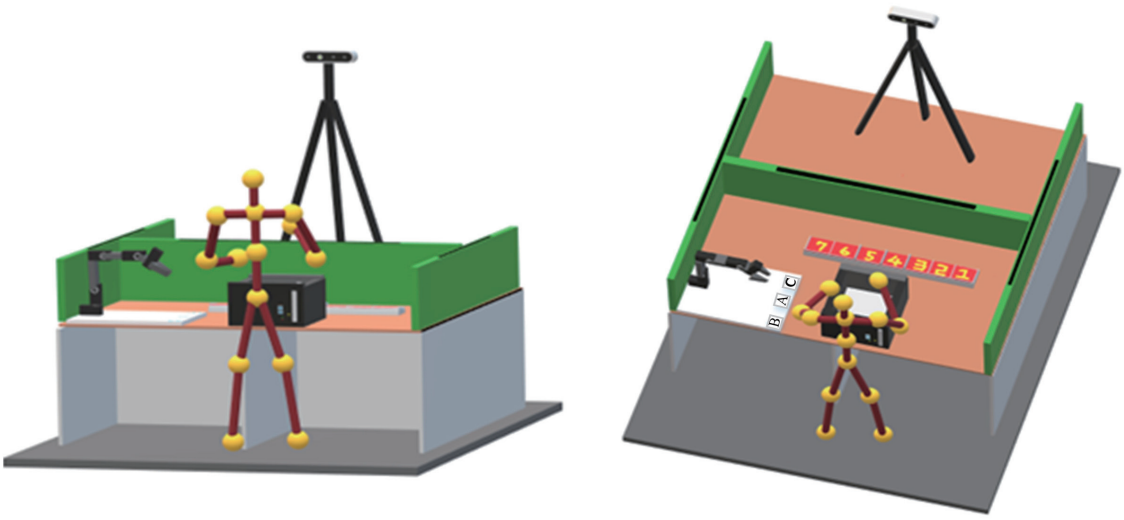


Fig. 1. (Color online) Practical application scenario of the presented HRC-based assembly line where the RGB-D image sensor is deployed to capture operator gestures for further recognition.

Table 1
Interaction list between the operator and the manipulator robot in an HRC task.

Interaction case	Human gesture action	Robot feedback action
Case 1	Acquiring (the object from the robot)	Opening the gripper and releasing the grabbed object to the hand of the operator
Case 2	Winding (the tape)	Grabbing the object in position A and moving to the operator side (scissors)
Case 3	Pulling (the cable tie)	Grabbing the object in position A and moving to the operator side (pincers)
Case 4	Picking up the item in position 1	Grabbing the object in position A and moving to the operator side
Case 5	Picking up the item in position 2	Grabbing the object in position B and moving to the operator side
Case 6	Picking up the item in position 3	Grabbing the object in position B and moving to the operator side
Case 7	Picking up the item in position 4	Grabbing the object in position B and moving to the operator side
Case 8	Picking up the item in position 5	Grabbing the object in position B and moving to the operator side
Case 9	Picking up the item in position 6	Grabbing the object in position C and moving to the operator side
Case 10	Picking up the item in position 7	Grabbing the object in position C and moving to the operator side

seven different locations). Each operator gesture action will be reacted by an associated robot action. As shown in Fig. 1, an image sensor (the RGB-D sensor employed in this study) is utilized by "ceiling mount-like" deployment to finely capture the active gesture data of the operator. The captured gesture data of the operator, mainly containing three different sensing modalities of data, namely, the RGB image data, the depth image data, and the skeleton-(x, y, z) raw data, are then classified (ten significant gesture action categories in total listed in Table 1 as

mentioned) using the developed gesture recognition scheme. The gesture recognition presented in this study will be designed as the manner of decision fusion of three data modalities of deep learning recognition channels, which will be described in detail in Sect. 3. Depending on the recognized operator gesture categorization, the manipulator will then perform the feedback action (i.e., the corresponding reaction). In the task completion by the proposed HRC-based assembly line, the manipulator will be viewed as a "professional assistant with the machine vision," which can discriminate the operator's significant actions and then provide the immediate service to the operator (e.g., delivering the required tool to the operator instantly). In Case 3 in Table 1, wherein the operator makes the gesture action of 'pulling' (pulling the cable tie), the manipulator will grab the object in position A (the tool of pincers) and then move to the operator side. In Case 1 wherein the manipulator finishes moving the grabbed object to the side of the operator, the operator will make the gesture action of 'acquiring.' After the constructed gesture recognition system correctly classifies such an action of acquiring the object from the robot, the corresponding feedback action of the manipulator is then to open the gripper to release the grabbed object, and the object will finally fall on the hand of the operator. The presented HRC-based assembly line with a cooperator of the smart manipulator assistant that can understand the assembly action of the operator and then immediately provide assistance during assembly tasks will further promote the automation of current factories or manufacturing fields.

3. Operator Gesture Recognition by Decision Fusion of Deep Learning Recognition Channels of Three Different Sensing Modalities of Gesture Data

Figure 2 shows the recognition calculation flowchart of gestures performed by the operator. Note that a detection scheme of the waking-up gesture of the operator is also designed in this system to be able to finely "wake up" the gesture recognition system to start to acquire the significant gesture action (one of the ten operation gesture cases, Case 1, Case 2, …, and Case 10, defined in the system, as mentioned previously) from the operator. As shown in Fig. 2, when the recognition scheme is triggered by the detection scheme, three different modalities of continuous-time sensing data, namely, 3D-(x, y, z) raw data, RGB images, and depth images, will be sent to three corresponding separated deep learning recognition computation channels, "3D-(x, y, z) raw LSTM," "RGB CNN-LSTM," and "depth CNN-LSTM," respectively. The decision

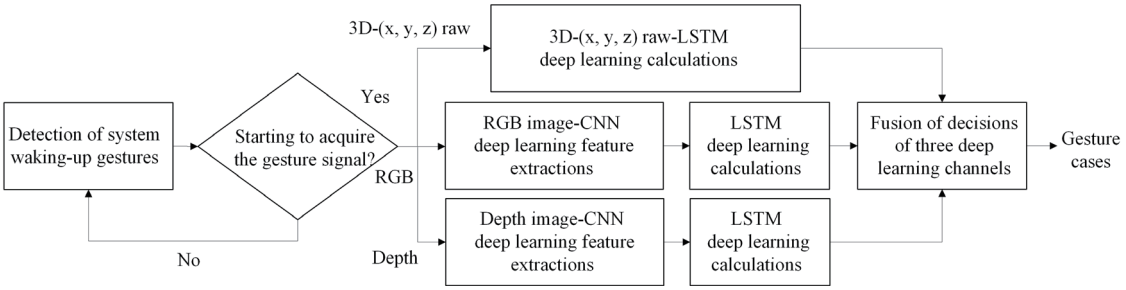


Fig. 2. Decision fusion of three separated deep learning recognition channels, each of which has a different sensing data modality.

fusion is finally designed to combine all determination outputs of three separated deep learning recognition channels. The CNN used in this work belongs to the type of VGGNet (mainly the configuration of 16 layers, the VGG-16 structure), and such a category of the CNN model is sometimes also called VGG16-CNN.[18] The LSTM in all constructed deep learning recognition channels is a standard recurrent neural network (RNN) and is composed of a series of LSTM units, each of which has a cell, an input gate, an output gate, and a forget gate.[19]

Figure 3 shows the image sensor, the Intel RealSense RGB-D image capture device, employed in this work. With fine releases of the 3D software development kit (SDK) from Intel, the 3D-skeleon information of the operator can then be obtained. The captured 3D skeleton from the Intel sensing device is mainly composed of 15 kernel joints [see Fig. 4(a)], "Joint-1: head," "Joint-2: neck," "Joint-3: left shoulder," "Joint-4: right shoulder," "Joint-5: left elbow," "Joint-6: right elbow," "Joint-7: left hand," "Joint-8: right hand," "Joint-9: torso," "Joint-10: left hip," "Joint-11: right hip," "Joint-12: left knee," "Joint-13: right knee," "Joint-14: left root," and "Joint-15: right root." Note that in the practical assembly-line operation scenario, space location values of nine joints, Joint-1 to Joint-9, are apparently variant when a specific operation gesture is made by the operator. Only these nine joints are considered to extract the (x, y, z)-coordinate

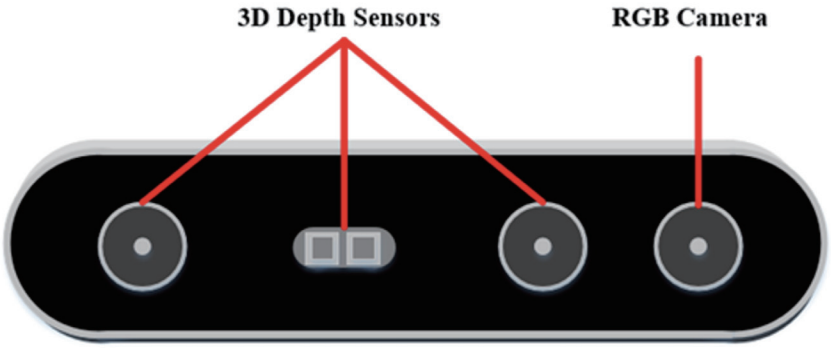


Fig. 3. (Color online) Intel RealSense RGB-D image capture device employed in this work for gesture data acquisitions of 3D-(x, y, z) raw data, RGB images, and depth images.

(a) (b) (c)

Fig. 4. (Color online) Operator gesture data captured from the Intel RGB-D sensing device with three different sensing modalities: (a) 3D-(x, y, z) raw data, (b) RGB images, and (c) depth images.

values. In this work, the 3D-(x, y, z) raw data has 27 dimensions, including three spatial dimensions of the x-, y-, and z-axes, contained in each of these nine joints [see Fig. 4(b)]. Figure 4(c) represents the gesture image with the depth type captured from IR transmitters and receivers of the RGB-D device.

Note that in the system wake-up scheme, the wake-up gesture of the operator is designed as the gesture of "both palms put together (or much close)." The detection of such a wake-up gesture is conceptually simple and computationally inexpensive, mainly concerning variations of (x, y, z)-coordinate values of two joints, Joint-7 and Joint-8. The term $Diff_{palm\_joint}$ defined in Eq. (1) can finely describe such coordinate value variations:

$$Diff_{palm\_joints} = \sqrt{\left(x_{Joint\text{-}7} - x_{Joint\text{-}8}\right)^2 + \left(y_{Joint\text{-}7} - y_{Joint\text{-}8}\right)^2 + \left(z_{Joint\text{-}7} - z_{Joint\text{-}8}\right)^2}. \quad (1)$$

Note that the calculated $Diff_{palm\_joint}$ is essentially the Euclidean distance between Joint-7 and Joint-8. When $Diff_{palm\_joint}$ is extremely small or approaches 0, a wake-up gesture action will then be detected by the system. Figure 5 illustrates a series of gesture actions performed by the operator, beginning at a wake-up gesture (T0), followed by the gesture of picking up the item (T1 and T2), another wake-up gesture (T5), and the gesture of acquiring the object from the robot (T6). In Fig. 5, the time points T3 and T4 denote the robot feedback action, grabbing the object in a specific position and moving to the operator side; the time point T7 is the robot feedback action of the gesture action of Case 1, which is to open the robot gripper and then release the grabbed object to the hand of the operator.

In the design of operator assembly-line gesture action recognition (as mentioned previously, a total of ten defined gesture classes are categorized by the system), the recognition decisions finally obtained from each of these three separated recognition channels, "3D-(x, y, z) raw LSTM," "RGB CNN-LSTM," and "depth CNN-LSTM," will be combined by weight combinations of soft-max outputs (see Figs. 6–8).

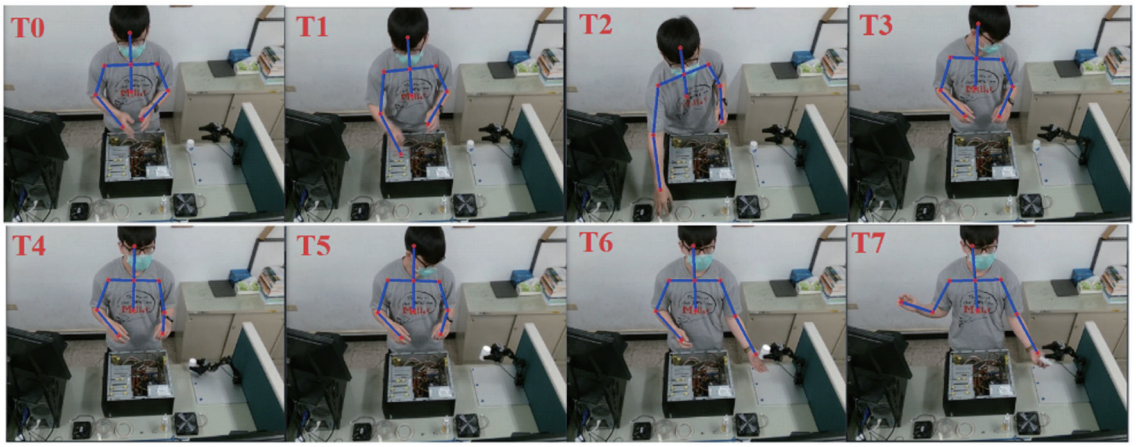


Fig. 5. (Color online) Series of operator gesture actions with the wake-up gesture of triggering the recognition system followed by the corresponding robot feedback action.

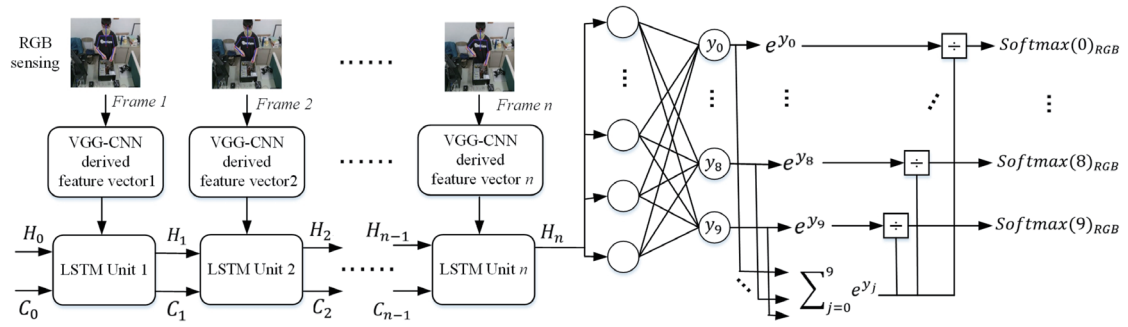


Fig. 6. (Color online) Recognition channel to process operator gesture data with the RGB sensing modality (RGB CNN-LSTM channel).

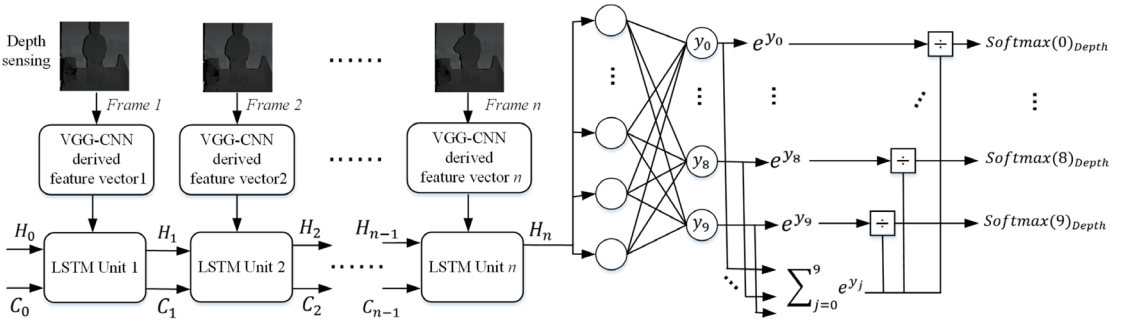


Fig. 7. (Color online) Recognition channel to process operator gesture data with the depth sensing modality (depth CNN-LSTM channel).

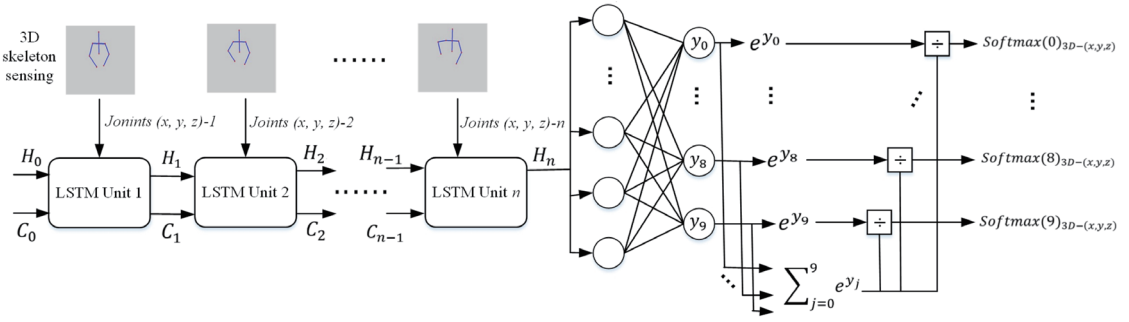


Fig. 8. (Color online) Recognition channel to process operator gesture data with the 3D-(x, y, z) skeleton sensing modality [3D-(x, y, z) raw LSTM channel].

As mentioned above, in the presented HRC assembly-line system, a total of ten operator gesture categories should be recognized, and therefore, the recognition decision set derived from each of the three separated recognition channels will have ten calculated output scores. Each of the ten calculated output scores is related to the corresponding gesture category. As shown in Fig. 6, after the recognition system is triggered by the wake-up gesture of the operator, a continuous-time assembly operation gesture data with n RGB-type frames made by the operator

is sent to the recognition channel of RGB CNN-LSTM. When the channel finishes recognition calculations to this input RGB image set, ten decision output scores can then be estimated from the final classification computation of the soft-max, which are $Softmax(0)_{RGB}$, $Softmax(1)_{RGB}$, …, and $Softmax(9)_{RGB}$. Accompanied by the RGB-type gesture data, the other two modalities of data, namely, the depth and 3D-(x, y, z) raw sensing data, will also be acquired from the RGB-D image capture device. As recognition decisions of the final output of the RGB CNN-LSTM channel, ten decision output scores of $Softmax(0)_{Depth}$, $Softmax(1)_{Depth}$, …, and $Softmax(9)_{Depth}$, and another ten decision output scores of $Softmax(0)_{3D\text{-}(x,y,z)}$, $Softmax(1)_{3D\text{-}(x,y,z)}$, …, and $Softmax(9)_{3D\text{-}(x,y,z)}$ will also be obtained from the final soft-max classification computation of recognition channels of the depth CNN-LSTM and 3D-(x, y, z) raw LSTM, respectively (see Figs. 7 and 8). For the acquired input gesture data of the operator, the weight combination scheme to simultaneously take into account these three different sensing modalities of soft-max score output sets to finally make the recognition result is designed as follows [Eqs. (2–4)]:

$$Softmax(i)_{Mixed} = w_{RGB} \cdot Softmax(i)_{RGB} + w_{Depth} \cdot Softmax(i)_{Depth} + w_{3D} \cdot Softmax(i)_{3D\text{-}(x,y,z)}, \quad i = 0, 1, \ldots, 9, \tag{2}$$

$$w_{RGB} + w_{Depth} + w_{3D} = 1, \tag{3}$$

$$Recognized\_Label = \underset{j \in \{0,1,\ldots 9\}}{\operatorname{argmax}} Softmax(j)_{Mixed}. \tag{4}$$

Note that in Eq. (3), there will be infinite conditions on the weight set $\{w_{RGB}, w_{Depth}, w_{3D}\}$, which will be an annoying trial-and-test problem in the practical application scenario. In this study, we provide three different weight allocation strategies to determine such a weight set, which are "high-low-slight," "half-quarter-quarter," and "weighted-average." In the high-low-slight weight allocation strategy, one of the three different gesture sensing modalities will be given the highest weight (more than 0.5), denoting the relatively higher confidence in the recognition result of the specific sensing modality. In the half-quarter-quarter weight allocation strategy, a weight of 0.5 will be set to one specific sensing modality, and the remaining two modalities will be equally set the same weight of 0.25. As for the weighted-average weight allocation strategy, three sensing modalities will be treated completely the same, that is, they will be given similar weights.

4. Experiments

Experiments of the proposed assembly-line design using HRCs with operator gesture recognition are conducted in a laboratory office environment. The operator gesture database that contains ten different categories of computer device assembly gestures is established by three male persons. As mentioned earlier, the image capture sensor is the Intel RealSense sensing device, belonging to the RGB-D type of image acquisition system. The top-to-down sensor

deployment is used to capture the gesture data of the operator. Each of the three male persons is requested to make 200 gesture actions, i.e., 20 actions collected for each of the ten different categories of assembly operation gestures. The time period of each gesture action acquired from the sensor device is set as 1 s. A computer device assembly mission is arranged in this work where these ten defined operator gesture types are (1) to receive the object released from the manipulator, (2) to wind the cable tie to bind the bus or wind the cupper line, (3) to pull up the assembled object to remove it from the device, (4) to pick up the item (computer fan) in location-1, (5) to pick up the item (superglue) in location-2, (6) to pick up the item (small screw) in location-3, (7) to pick up the item (large screw) in location-4, (8) to pick up the item (double-sided tape) in location-5, (9) to pick up the item (seal tape) in location-6, and (10) to pick up the item (CPU fan) in location-7. The established operator gesture database containing a total of 600 gesture actions (three different image sensing modalities, namely, the RGB image, the depth image, and the 3D skeleton raw data, included in each time of captured gesture actions, as shown in Table 2) is divided into two parts: the first half for the model training of various deep learning gesture recognition systems and the other half for performance evaluations of all constructed gesture recognition models. A personal computer (PC) with Windows 10, CPU of Intel Xeon

Table 2
(Color online) Continuous-time RGB, depth, and 3D-(x, y, z) sensing data acquired simultaneously from the RGB-D device in each of ten different cases of operator assembly gesture actions.

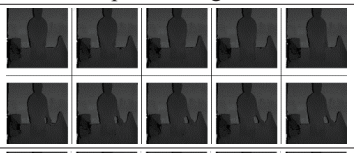
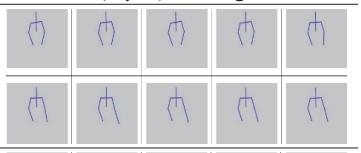
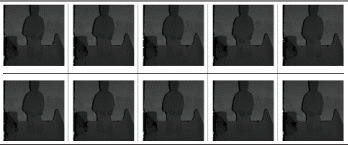
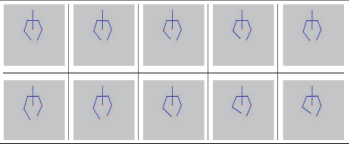
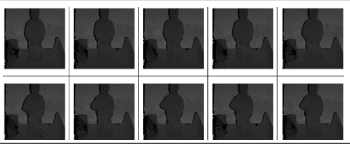
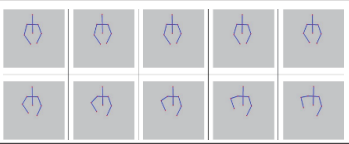
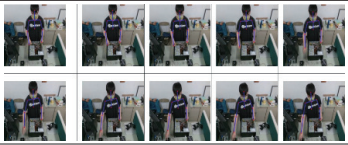
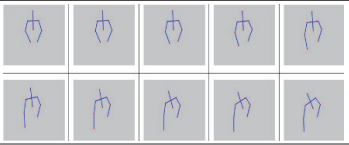
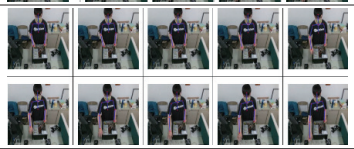
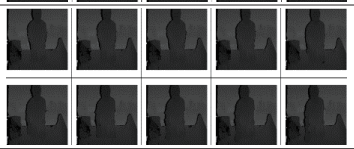
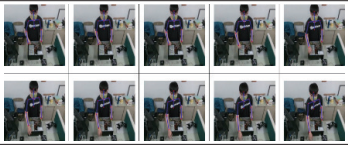
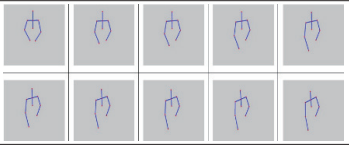
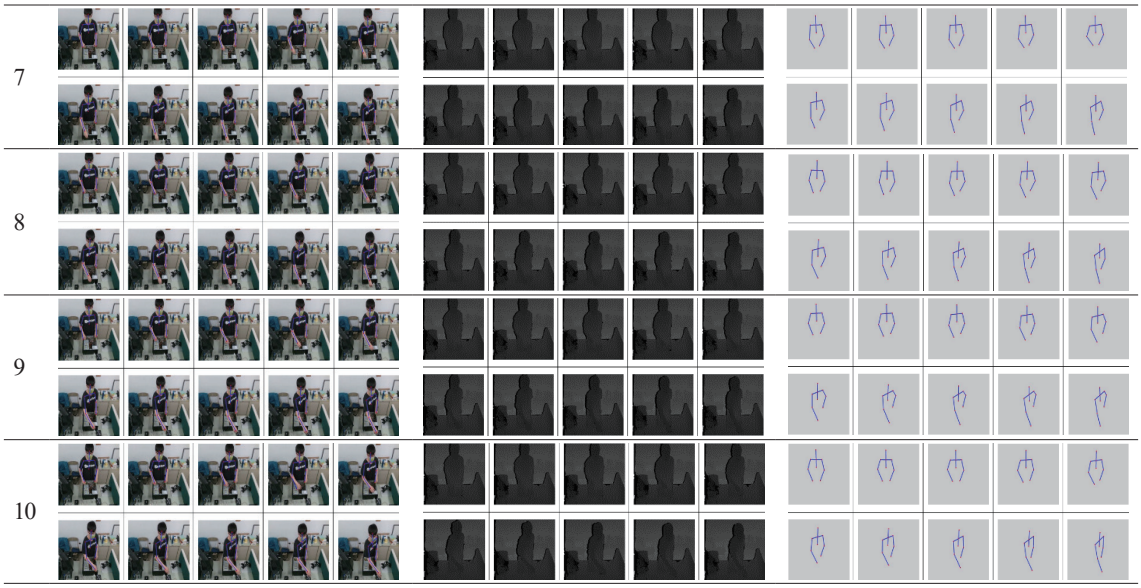
	RGB sensing data	Depth sensing data	3D-(x, y, z) sensing data
1			
2			
3			
4			
5			
6			

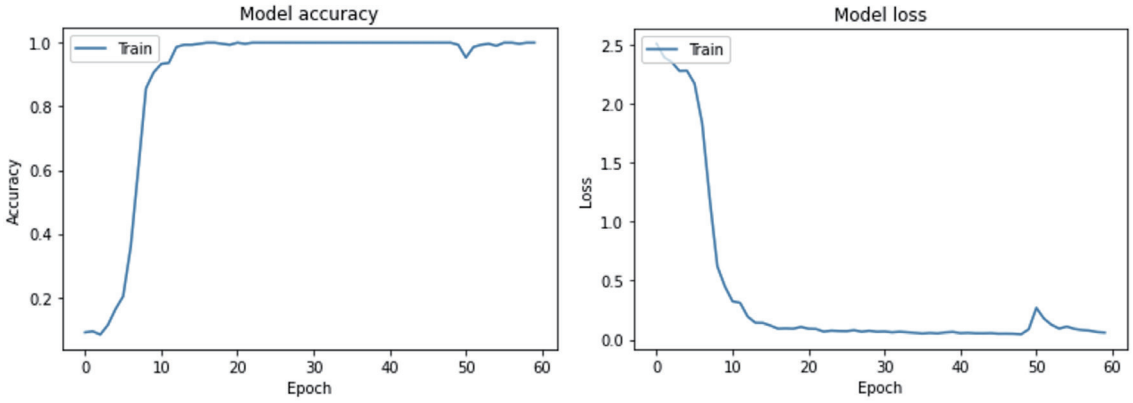
Table 2
(Color online) (Continued) Continuous-time RGB, depth, and 3D-(x, y, z) sensing data acquired simultaneously from the RGB-D device in each of ten different cases of operator assembly gesture actions.



W-2235, RAM of 32 GB, and GPU of NVIDIA GeForce RTX3080ti is used in this work to perform all calculation tasks in both the test and training phases of deep learning gesture recognition. The PC is connected to OpenMANIPULATOR-X via the OpenCR embedded board for data communication.[20]

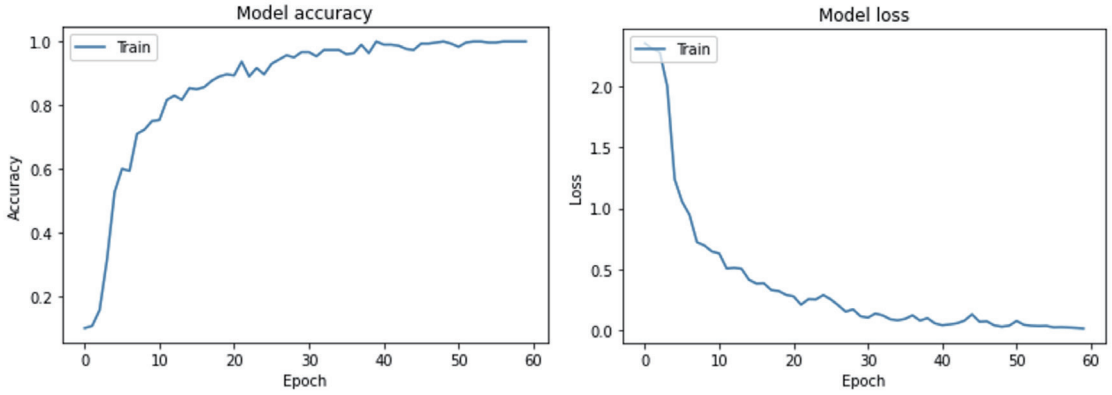
Figures 9–11 depict the performance curves (mainly the recognition accuracy and model loss rate curves) of deep learning models of RGB CNN-LSTM, depth CNN-LSTM, and 3D-(x, y, z) raw LSTM recognition channels, respectively, in the training phase. As shown in both the recognition accuracy and model loss rate curves, each of the three different sensor modalities of recognition channels can achieve a fine deep learning training calculation at the final iterative training procedure and satisfactory model performance with the recognition rate approaching 100% and the loss rate approaching 0%.

In the testing phase of the HRC assembly-line design with operator gesture recognition, each type of separated deep learning channel is first evaluated in terms of recognition performance. Table 3 shows the operator gesture recognition accuracies obtained using the RGB CNN-LSTM channel alone, the depth CNN-LSTM channel alone, and the 3D-(x, y, z) raw LSTM channel alone. As shown in Table 3, in the recognition of ten different categories of operator assembly gestures, the RGB CNN-LSTM recognition channel apparently has the highest recognition rate of 95%, which is followed by 87.66% for the depth CNN-LSTM recognition channel, and the performance of the 3D-(x, y, z) raw LSTM recognition channel is extremely close to that of the depth CNN-LSTM channel, reaching the recognition rate of 87.33%. Table 4 shows the confusion matrix of gesture recognition by the RGB CNN-LSTM channel alone. Tables 5–7 show the gesture recognition rates of weight combination schemes of high-low-slight, weighted-average, and half-quarter-quarter, respectively, on the decision fusion of three different sensor



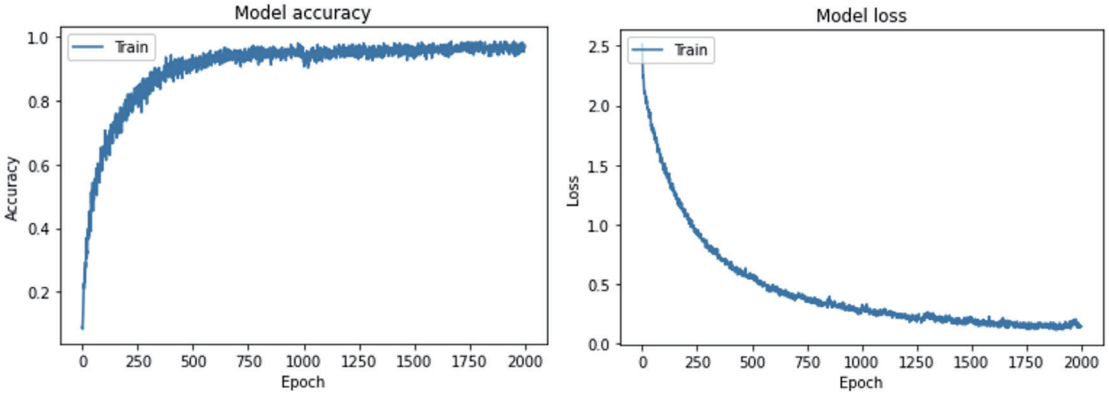
(a) (b)

Fig. 9. (Color online) (a) Accuracy rate curve and (b) loss rate curve of the recognition channel of RGB CNN-LSTM deep learning in the training phase of operator gesture recognition.



(a) (b)

Fig. 10. (Color online) (a) Accuracy rate curve and (b) loss rate curve of the recognition channel of depth CNN-LSTM deep learning in the training phase of operator gesture recognition.



(a) (b)

Fig. 11. (Color online) (a) Accuracy rate curve and (b) loss rate curve of the recognition channel of 3D-(x, y, z) raw LSTM deep learning in the training phase of operator gesture recognition.

Table 3
Operator assembly gesture recognition accuracies obtained using the separated deep learning recognition channels of RGB CNN-LSTM, depth CNN-LSTM, and 3D-(x, y, z) raw LSTM.

Recognition channel with only one modality	Recognition accuracy (%)
RGB CNN-LSTM channel (RGB data only)	95
Depth RGB CNN-LSTM channel (depth data only)	87.66
3D-(x, y, z) raw LSTM channel (3D skeleton data only)	87.33

Table 4
Confusion matrix of operator assembly gesture recognition by the separated deep learning recognition channel of RGB CNN-LSTM.

	1	2	3	4	5	6	7	8	9	10
1	30	0	0	0	0	0	0	0	0	0
2	0	30	0	0	0	0	0	0	0	0
3	0	0	30	0	0	0	0	0	0	0
4	0	0	0	29	1	0	0	0	0	0
5	0	0	0	0	30	0	0	0	0	0
6	0	0	0	0	0	25	5	0	0	0
7	0	0	0	0	0	4	23	3	0	0
8	0	0	0	0	0	0	0	29	1	0
9	0	0	0	0	0	0	0	0	29	1
10	0	0	0	0	0	0	0	0	0	30

Table 5
High-low-slight weight combination scheme used to achieve decision fusion among gesture recognition channels of RGB, depth, and 3D image sensing modalities.

Condition	w_{RGB}	w_{Depth}	w_{3D}	Recognition accuracy (%)
1	0.6	0.3	0.1	95
2	0.6	0.1	0.3	95
3	0.3	0.6	0.1	89
4	0.1	0.6	0.3	88.66
5	0.3	0.1	0.6	91.33
6	0.1	0.3	0.6	92.00

Table 6
Weighted-average weight combination scheme used to achieve decision fusion among gesture recognition channels of RGB, depth, and 3D image sensing modalities.

Condition	w_{RGB}	w_{Depth}	w_{3D}	Recognition accuracy (%)
1	0.34	0.33	0.33	93.66

Table 7
Half-quarter-quarter weight combination scheme used to achieve decision fusion among gesture recognition channels of RGB, depth, and 3D image sensing modalities.

Condition	w_{RGB}	w_{Depth}	w_{3D}	Recognition accuracy (%)
1	0.5	0.25	0.25	96
2	0.25	0.5	0.25	89.66
3	0.25	0.25	0.5	93

Table 8
Confusion matrix of operator assembly gesture recognition by decision fusion of the half-quarter-quarter weight combination with (w_{RGB}, w_{Depth}, w_{3D}) = (0.5, 0.25, 0.25).

	1	2	3	4	5	6	7	8	9	10
1	30	0	0	0	0	0	0	0	0	0
2	0	30	0	0	0	0	0	0	0	0
3	0	0	30	0	0	0	0	0	0	0
4	0	0	0	29	1	0	0	0	0	0
5	0	0	0	0	30	0	0	0	0	0
6	0	0	0	0	0	27	3	0	0	0
7	0	0	0	0	0	5	23	2	0	0
8	0	0	0	0	0	0	0	29	1	0
9	0	0	0	0	0	0	0	0	30	0
10	0	0	0	0	0	0	0	0	0	30

modalities of RGB, depth, and 3D image data. The recognition outcomes in Table 5 reveal that the highest recognition performance of the high-low-slight weight combination appears when (w_{RGB}, w_{Depth}, w_{3D}) = (0.6, 0.3, 0.1) or (w_{RGB}, w_{Depth}, w_{3D}) = (0.6, 0.1, 0.3). In each of these two best cases, the recognition performance remains the same as that of the RGB CNN-LSTM recognition channel alone; for the weight combination scheme of weighted-average, such combination with the same weight set to achieve the decision fusion of three recognition channel outputs is ineffective, even more unideal than the RGB CNN-LSTM recognition channel alone (see Table 6); in all weight combination schemes, the half-quarter-quarter approach has the most satisfactory performance. As shown in Table 7, the highest recognition rate appears in the case of (w_{RGB}, w_{Depth}, w_{3D}) = (0.5, 0.25, 0.25), in which the operator gesture recognition rate can approach 96%, higher than 95% of the RGB CNN-LSTM channel alone. Table 8 shows the confusion matrix of recognition of ten operator assembly gesture classes by the half-quarter-quarter weight combination with the setting of (w_{RGB}, w_{Depth}, w_{3D}) = (0.5, 0.25, 0.25) on the decision fusion of RGB, depth, and 3D raw recognition channel outputs. Recognition experiment results demonstrate the effectiveness of the use of decision fusion in hybridizations of deep learning recognition channels of the three image sensing modalities of RGB, depth, and 3D raw data.

5. Conclusions

In this work, a smart assembly-line design by HRCs with the recognition of operator assembly gestures is proposed. In the HRC-based assembly task, the operator and manipulator will cooperate in an efficient manner where the manipulator will perform the feedback action according to the recognized operator gestures. A decision fusion design to hybridize all calculation outputs of deep learning recognition channels of three different image sensing modalities, namely, RGB, depth, and 3D-(x, y, z) raw, is presented in this study to increase the recognition accuracy of gesture recognition by only a separated deep learning recognition channel with the specific image sensing type. The developed system with incorporations of HRC and operator gesture recognition can further upgrade the assembly-line task in smart

manufacturing. On the basis of the system proposed in this study, a possible extension work can be further explored to improve the HRC mode, i.e., to promote the role of the manipulator robot, from being an assistant performing only tool delivery to the more collaborative co-worker that can complete parts of the assembly task independently with understanding of the task progress.

Acknowledgments

We acknowledge the support given by the National Science and Technology Council (NSTC), Taiwan under Grant No. 111-2221-E-239-033.

References

1 I. J. Ding, M. C. Hsieh, and Z. X. Wu: Sens. Mater. **35** (2023) 1099. https://doi.org/10.18494/SAM4236
2 F. Chen, Q. Zhong, F. Cannella, K. Sekiyama, and T. Fukuda: Int. J. Adv. Robot. Syst. **12** (2015). https://doi.org/10.5772/60044
3 B. Sadrfaridpour, H. Saeidi, and Y. Wang: Proc. IEEE Int. Conf. Automation Science and Engineering (IEEE, 2016) 462. https://doi.org/10.1109/COASE.2016.7743441
4 R. Bejarano, B. R. Ferrer, W. M. Mohammed, and J. L. Martinez Lastra: Proc. IEEE 17th Int. Conf. Industrial Informatics (IEEE, 2019) 557. https://doi.org/10.1109/INDIN41052.2019.8972158
5 N. Dimitropoulos, T. Togias, N. Zacharaki, G. Michalos, and S. Makris: Appl. Sci. **11** (2021) 5699. https://doi.org/10.3390/app11125699
6 E. Coupeté, F. Moutarde, and S. Manitsaris: Procedia Manuf. **3** (2015) 518. https://doi.org/10.1016/j.promfg.2015.07.216
7 A. Sylari, B. R. Ferrer, and J. L. M. Lastra: Proc. 2019 IEEE 17th Int. Conf. Industrial Informatics (IEEE, 2019) 827. https://doi.org/10.1109/INDIN41052.2019.8972301
8 Y. Cheng, L. Sun, C. Liu, and M. Tomizuka: IEEE Rob. Autom. Lett. **5** (2020) 2602. https://doi.org/10.1109/LRA.2020.2972874
9 G. Canal, S. Escalera, and C. Angulo: Comput. Vision Image Understanding **149** (2016) 65. https://doi.org/10.1016/j.cviu.2016.03.004
10 C. S. Lin, P. C. Chen, Y. C. Pan, C. M. Chang, and K. L. Huang: J. Sens. **2020** (2020) 9270829. https://doi.org/10.1155/2020/9270829
11 E. Coupeté, F. Moutarde, and S. Manitsaris: Auton. Rob. **43** (2019) 1309. https://doi.org/10.1007/s10514-018-9704-y
12 N. Dimitropoulos, T. Togias, G. Michalos, and S. Makris: Procedia CIRP **97** (2021) 464. https://doi.org/10.1016/j.procir.2020.07.006
13 I. J. Ding and Y. C. Juang: Sens. Mater. **34** (2022) 3513. https://doi.org/10.18494/SAM4045
14 I. J. Ding and N. W. Zheng: Sensors **22** (2022) 803. https://doi.org/10.3390/s22030803
15 I. J. Ding and N. W. Zheng: Sens. Mater. **34** (2022) 203. https://doi.org/10.18494/SAM3557
16 I. J. Ding, N. W. Zheng, and M. C. Hsieh: J. Intell. Fuzzy Syst. **40** (2021) 7775. https://doi.org/10.3233/JIFS-189598
17 OpenMANIPULATOR-X. https://www.roscomponents.com/en/robotic-arms/openmanipulator-x
18 K. Simonyan and A. Zisserman: Proc. Int. Conf. Learning Representations (2015). https://arxiv.org/abs/1409.1556
19 S. Hochreiter and J. Schmidhuber: Neural Comput. **9** (1997) 1735. https://doi.org/10.1162/neco.1997.9.8.1735
20 OpenCR. https://robots.ros.org/opencr/